comment-installer-prestashop-sur-ubuntu-22-04

Prestashop is a free and open-source e-commerce solution written in PHP. It allows you to self-hosted and create online stores and grow your online business. Prestashop is a fully-customizable and feature-rich e-commerce solution for building comprehensive eCommerce websites. In the meantime, Prestashop is used by more than 250.000 online stores around the globe and is available in 65 languages.

In this tutorial, we are going to explain how to install an open-source eCommerce solution - Prestashop using the LAMP Stack (Linux, Apache, MySQL/MariaDB, and PHP) on the Ubuntu 22.04 server. This tutorial also includes the basic configuations of LAMP Stack for PHP web applications.

Prerequisite

Here's what you need to complete this tutorial:

- An Ubuntu 22.04 server - This tutorial uses an ubuntu server with the hostname 'server-ubuntu' and the IP address is '192.168.5.100'.
- A non-root user with root administrative privileges.
- A domain name pointed to the Ubuntu Server IP address.

Installing Apache Web Server

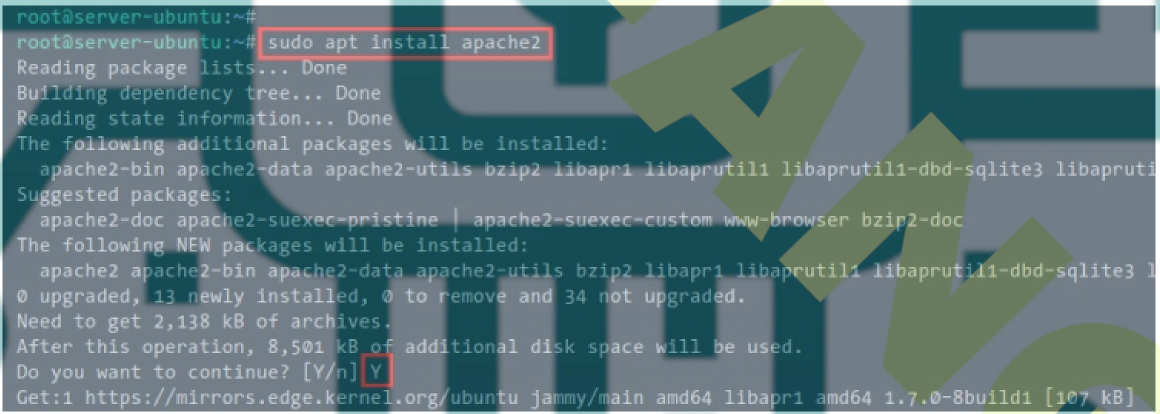
In the following step, you will install the Apache web server on your Ubuntu system. The Prestashop eCommerce requires at least the Apache web server **v2.2**. For this installation, you will install Apache **v2.4**, which is available by default on the Ubuntu repositories.

We always recommend updating and refreshing your repositories before installing any packages. So, run the apt command to update Ubuntu repositories.

```
sudo apt update
```

Install the Apache web server using the following apt command. When prompted to confirm the installation, input **Y** to accept and press **ENTER**.

```
sudo apt install apache2
```

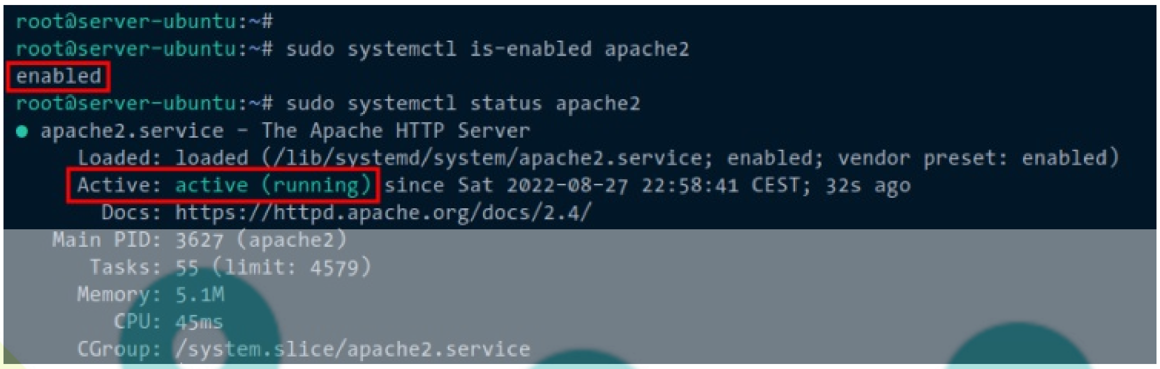


```
root@server-ubuntu:~#
root@server-ubuntu:~# sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils bzip2 libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libapruti
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser bzip2-doc
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils bzip2 libapr1 libaprutil1 libaprutil1-dbd-sqlite3 l
0 upgraded, 13 newly installed, 0 to remove and 34 not upgraded.
Need to get 2,138 kB of archives.
After this operation, 8,501 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 https://mirrors.edge.kernel.org/ubuntu jammy/main amd64 libapr1 amd64 1.7.0-8build1 [107 kB]
```

Once the Apache web server is installed, you need to check and verify the '**apache2**' service and make sure it's running. You can use the following systemctl command to check and verify the '**apache2**' service.

You will receive an output about the '**apache2**' service **enabled**, which means it will be running automatically at system startup. And the current status of the '**apache2**' service is **running**.

```
sudo systemctl is-enabled apache2
sudo systemctl status apache2
```

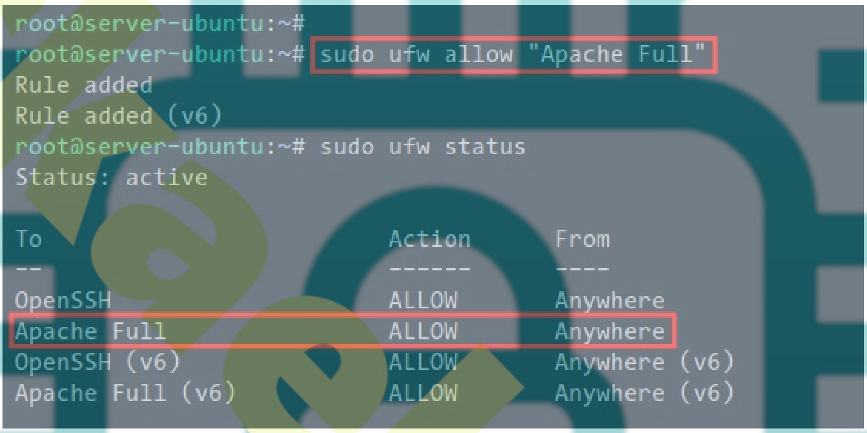


```
root@server-ubuntu:~#
root@server-ubuntu:~# sudo systemctl is-enabled apache2
enabled
root@server-ubuntu:~# sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
     Active: active (running) since Sat 2022-08-27 22:58:41 CEST; 32s ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 3627 (apache2)
      Tasks: 55 (limit: 4579)
     Memory: 5.1M
        CPU: 45ms
     CGroup: /system.slice/apache2.service
```

Lastly, you also need to add the HTTP and HTTPS ports to the UFW firewall. Use the following ufw command to add the custom rule "**Apache Full**" to the UFW firewall. Then, verify the list of enabled UFW rules.

You will receive an output that the rule "**Apache Full**" is added to the UFW firewall.

```
sudo ufw allow "Apache Full"
sudo ufw status
```



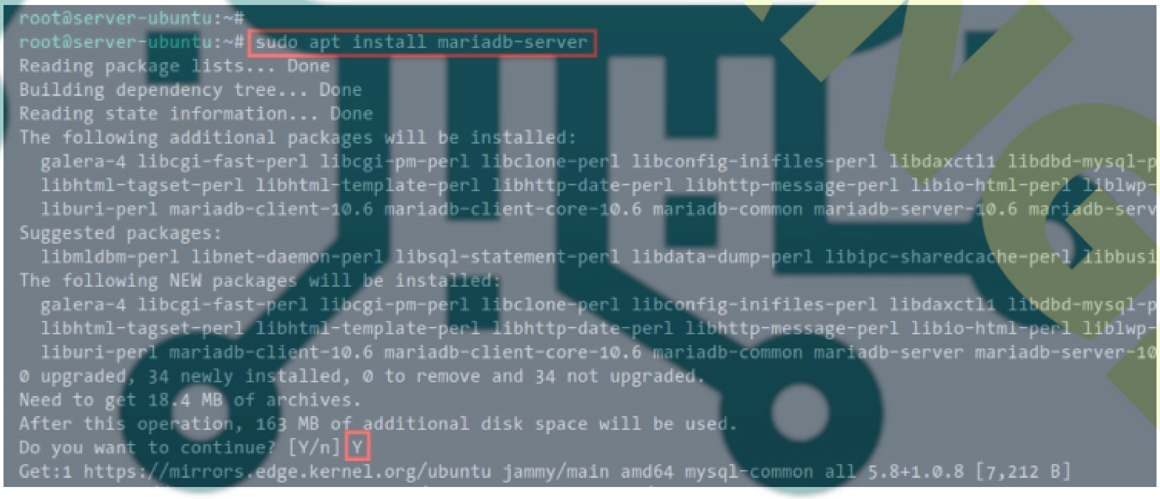
```
root@server-ubuntu:~#
root@server-ubuntu:~# sudo ufw allow "Apache Full"
Rule added
Rule added (v6)
root@server-ubuntu:~# sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
OpenSSH                    ALLOW       Anywhere
Apache Full                ALLOW       Anywhere
OpenSSH (v6)               ALLOW       Anywhere (v6)
Apache Full (v6)           ALLOW       Anywhere (v6)
```

Installing MariaDB Database Server

Since Prestashop only supports the MySQL/MariaDB database, you will now install this database server on your Ubuntu machine. In this example, you will install and use MariaDB as the database for your Prestashop installation.

Use the following apt command to install the MariaDB database server. Input **Y** when prompted to confirm the installation, then press **ENTER**.

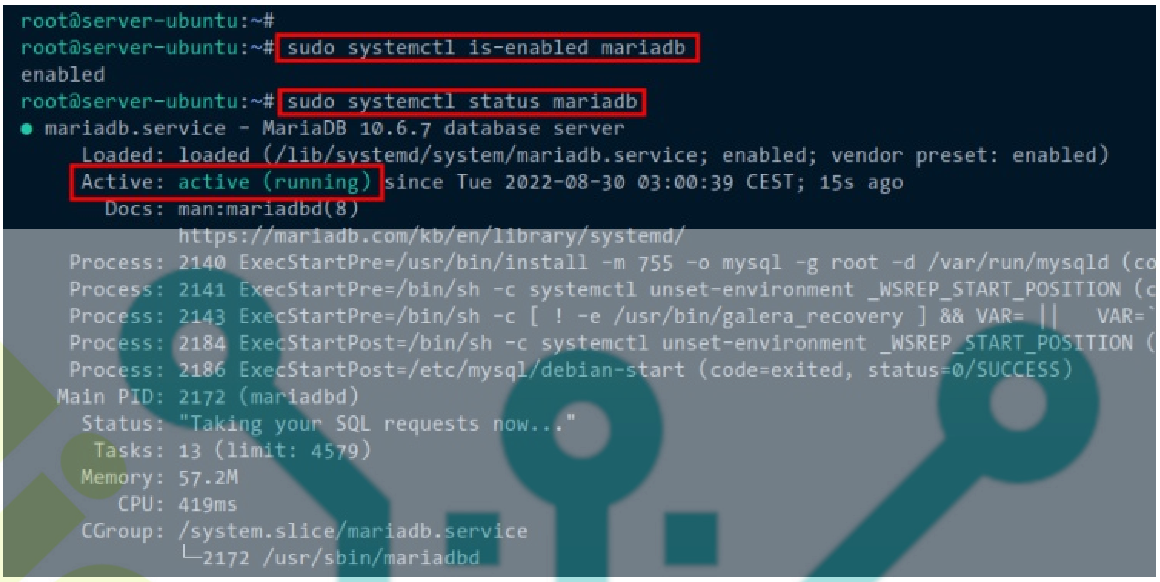
```
sudo apt install mariadb-server
```



```
root@server-ubuntu:~#
root@server-ubuntu:~# sudo apt install mariadb-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  galera-4 libcgi-fast-perl libcgi-pm-perl libclone-perl libconfig-inifiles-perl libdaxctl1 libdbd-mysql-p
  libhtml-tagset-perl libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl liblwp-
  liburi-perl mariadb-client-10.6 mariadb-client-core-10.6 mariadb-common mariadb-server-10.6 mariadb-serv
Suggested packages:
  libmldbm-perl libnet-daemon-perl libsql-statement-perl libdata-dump-perl libipc-sharedcache-perl libbusi
The following NEW packages will be installed:
  galera-4 libcgi-fast-perl libcgi-pm-perl libclone-perl libconfig-inifiles-perl libdaxctl1 libdbd-mysql-p
  libhtml-tagset-perl libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl liblwp-
  liburi-perl mariadb-client-10.6 mariadb-client-core-10.6 mariadb-common mariadb-server mariadb-server-10
0 upgraded, 34 newly installed, 0 to remove and 34 not upgraded.
Need to get 18.4 MB of archives.
After this operation, 163 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 https://mirrors.edge.kernel.org/ubuntu jammy/main amd64 mysql-common all 5.8+1.0.8 [7,212 B]
```

After the MariaDB server is installed, check and verify the '**mariadb**' service using the following systemctl command. You should receive an output that the '**mariadb**' service is **enabled**, which means it will be automatically running at system boot. And the status of the '**mariadb**' service is '**running**'.

```
sudo systemctl is-enabled mariadb
sudo systemctl status mariadb
```



```
root@server-ubuntu:~#
root@server-ubuntu:~# sudo systemctl is-enabled mariadb
enabled
root@server-ubuntu:~# sudo systemctl status mariadb
● mariadb.service - MariaDB 10.6.7 database server
     Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset: enabled)
     Active: active (running) since Tue 2022-08-30 03:00:39 CEST; 15s ago
       Docs: man:mariadbd(8)
             https://mariadb.com/kb/en/library/systemd/
    Process: 2140 ExecStartPre=/usr/bin/install -m 755 -o mysql -g root -d /var/run/mysqld (co
    Process: 2141 ExecStartPre=/bin/sh -c systemctl unset-environment _WSREP_START_POSITION (c
    Process: 2143 ExecStartPre=/bin/sh -c [ ! -e /usr/bin/galera_recovery ] && VAR= ||   VAR=`
    Process: 2184 ExecStartPost=/bin/sh -c systemctl unset-environment _WSREP_START_POSITION (
    Process: 2186 ExecStartPost=/etc/mysql/debian-start (code=exited, status=0/SUCCESS)
   Main PID: 2172 (mariadbd)
     Status: "Taking your SQL requests now..."
      Tasks: 13 (limit: 4579)
     Memory: 57.2M
        CPU: 419ms
     CGroup: /system.slice/mariadb.service
             └─2172 /usr/sbin/mariadbd
```

Lastly, you will also need to secure the MariaDB deployment. You can do this via the command-line '*mysql_secure_installation*' that is provided by MariaDB server packages.

Run the following '*mysql_secure_installation*' to start configuring and securing your MariaDB server deployment.

```
sudo mysql_secure_installation
```

You will be prompted with the following questions about the MariaDB server deployment:

- **Switch to unix_socket authentication**?. Input **n** and press **ENTER**. The default MariaDB root user is already protected. optionally, you can also enable it by typing **y** for yes.
- **Change the root password**?. Input **y** to confirm and set up your new MariaDB root password.
- R**emove anonymous user**?. Input **y** to confirm.
- **Disallow root login remotely**? Input **y** to confirm. Only local connection will be allowed if you are using the MariaDB root user.
- **Remove test database and access to it**?. Input **y** to confirm and remove the default database '**test**'.
- Lastly, input **y** again to **reload all tables privileges** on your MariaDB server and apply new changes.

Installing PHP

For the latest version of Prestashop installation, it's recommended to use at least PHP v7.1. In this demo, you will use **PHP 7.4** for the Prestashop, and you can install PHP 7.4 on the latest Ubuntu 22.04 system via third-party repository.

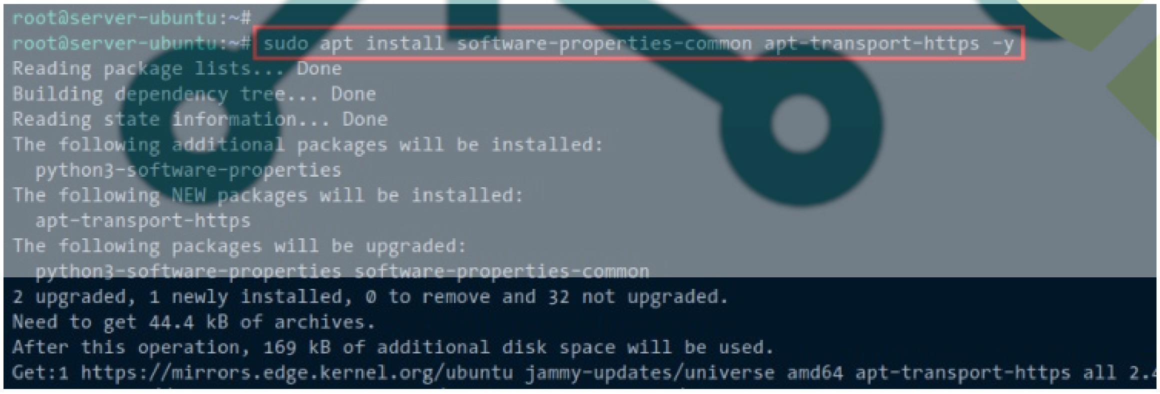
Before installing PHP, run the following apt command to install the basic packages dpendencies for managing repositories.

```
sudo apt install software-properties-common apt-transport-https -y
```

Picture

Now add the **PHP 7.4 PPA** repository to your Ubuntu system via the 'add-apt-repository' command below. Also, the following command will automatically update and refresh your Ubuntu repositories.

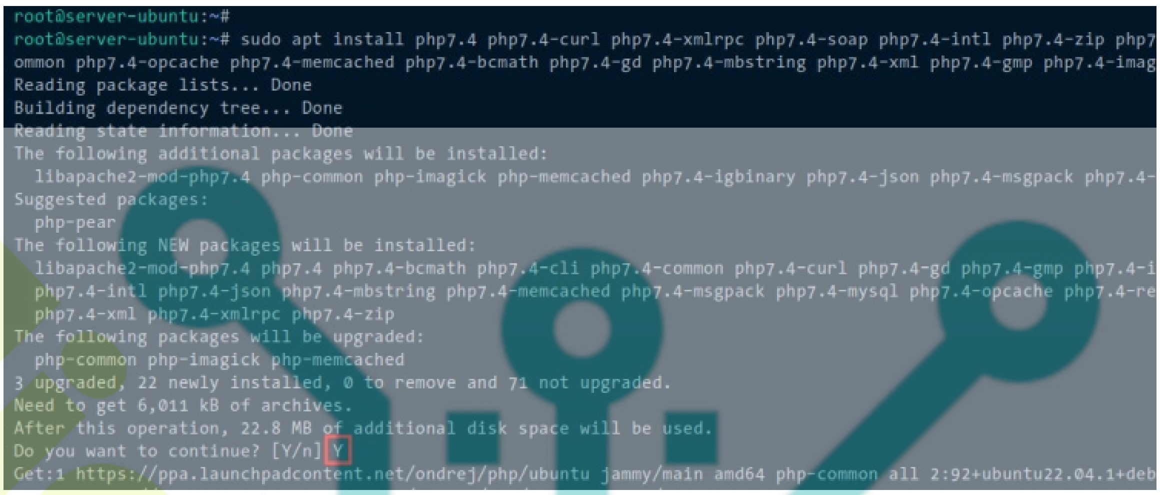
```
sudo add-apt-repository ppa:ondrej/php -y
```



```
root@server-ubuntu:~#
root@server-ubuntu:~# sudo apt install software-properties-common apt-transport-https -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  python3-software-properties
The following NEW packages will be installed:
  apt-transport-https
The following packages will be upgraded:
  python3-software-properties software-properties-common
2 upgraded, 1 newly installed, 0 to remove and 32 not upgraded.
Need to get 44.4 kB of archives.
After this operation, 169 kB of additional disk space will be used.
Get:1 https://mirrors.edge.kernel.org/ubuntu jammy-updates/universe amd64 apt-transport-https all 2.4
```

Next, use the following apt command to install PHP packages with some extensions for the Prestashop. When prompted to confirm the installation, input **Y** to agree and press **ENTER**.

```
sudo apt install php7.4 php7.4-curl php7.4-xmlrpc php7.4-soap php7.4-intl php7.4-zip php7.4-cli php7.4-mysql php7.4-common php7.4-opcache php7.4-memcached php7.4-bcmath php7.4-gd php7.4-mbstring php7.4-xml php7.4-gmp php7.4-imagick
```



```
root@server-ubuntu:~#
root@server-ubuntu:~# sudo apt install php7.4 php7.4-curl php7.4-xmlrpc php7.4-soap php7.4-intl php7.4-zip php7
ommon php7.4-opcache php7.4-memcached php7.4-bcmath php7.4-gd php7.4-mbstring php7.4-xml php7.4-gmp php7.4-imag
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libapache2-mod-php7.4 php-common php-imagick php-memcached php7.4-igbinary php7.4-json php7.4-msgpack php7.4-
Suggested packages:
  php-pear
The following NEW packages will be installed:
  libapache2-mod-php7.4 php7.4 php7.4-bcmath php7.4-cli php7.4-common php7.4-curl php7.4-gd php7.4-gmp php7.4-i
  php7.4-intl php7.4-json php7.4-mbstring php7.4-memcached php7.4-msgpack php7.4-mysql php7.4-opcache php7.4-re
  php7.4-xml php7.4-xmlrpc php7.4-zip
The following packages will be upgraded:
  php-common php-imagick php-memcached
3 upgraded, 22 newly installed, 0 to remove and 71 not upgraded.
Need to get 6,011 kB of archives.
After this operation, 22.8 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 https://ppa.launchpadcontent.net/ondrej/php/ubuntu jammy/main amd64 php-common all 2:92+ubuntu22.04.1+deb
```

After PHP is installed, open the PHP config file '*/etc/php/7.4/apache2/php.ini*' using your text editor. In this example, we will use nano.

```
sudo nano /etc/php/7.4/apache2/php.ini
```

Change details PHP configurations as below. Be sure to change the option '**date.timezone**' and '**memory_limit**' options with your current server environment.

```
date.timezone = Europe/Paris
max_execution_time = 130
memory_limit = 256M
allow_url_fopen = On
allow_url_include = Off
post_max_size = 128M
upload_max_filesize = 128M
max_input_vars = 5000
```

Save your changes and close the file when you are finished.

Next, run the following systemctl command below to restart the '**apache2**' service. Also, this will apply any changes to the PHP config file '**php.ini**'.

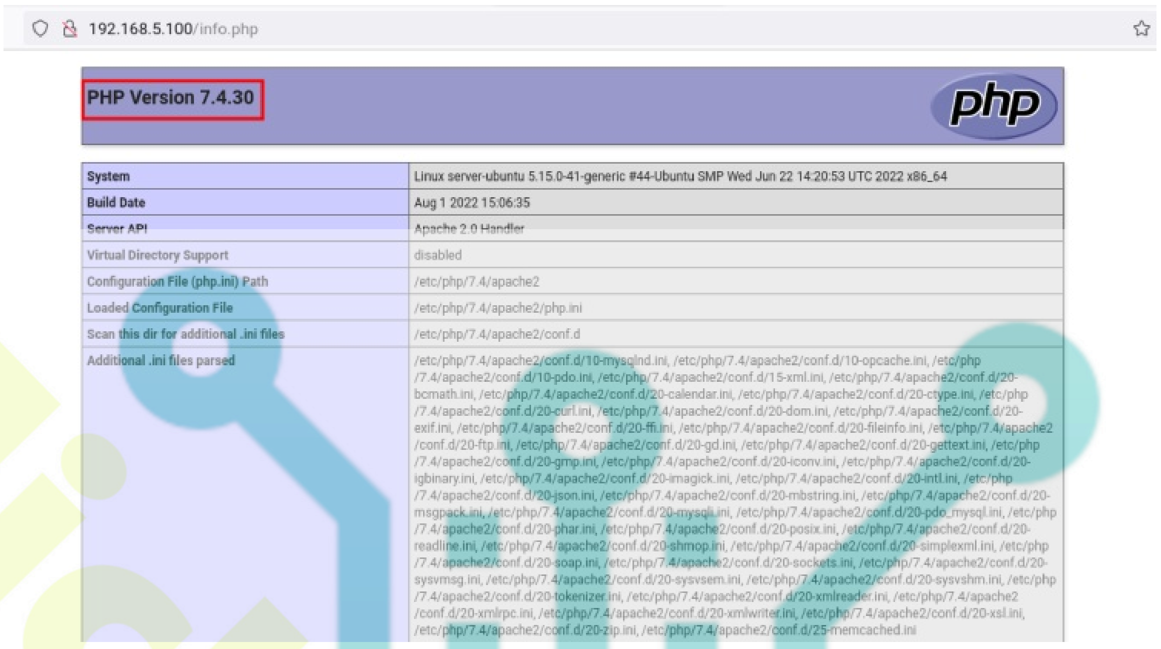
```
sudo systemctl restart apache2
```

Now the Apache web server and PHP should be running. You can verify that by creating the **phpinfo** file and testing it via the web browser.

Run the following command to create a new phpinfo file '*/var/www/html/info.php*'. This file should now be accessible via the URL path '*/info.ph*p'.

```
cat <<EOF | sudo tee /var/www/html/info.php
<?php
phpinfo();
?>
EOF
```

Open the web browser and access the phpinfo file via your server IP address followed by the path of the file '*/info.ph*p' (http://192.168.5.100/info.php). You should get a page with detailed information about your PHP installation.


192.168.5.100/info.php

PHP Version 7.4.30

System	Linux server-ubuntu 5.15.0-41-generic #44-Ubuntu SMP Wed Jun 22 14:20:53 UTC 2022 x86_64
Build Date	Aug 1 2022 15:06:35
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.4/apache2
Loaded Configuration File	/etc/php/7.4/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.4/apache2/conf.d
Additional .ini files parsed	/etc/php/7.4/apache2/conf.d/10-mysqlnd.ini, /etc/php/7.4/apache2/conf.d/10-opcache.ini, /etc/php/7.4/apache2/conf.d/10-pdo.ini, /etc/php/7.4/apache2/conf.d/15-xml.ini, /etc/php/7.4/apache2/conf.d/20-bcmath.ini, /etc/php/7.4/apache2/conf.d/20-calendar.ini, /etc/php/7.4/apache2/conf.d/20-ctype.ini, /etc/php/7.4/apache2/conf.d/20-curl.ini, /etc/php/7.4/apache2/conf.d/20-dom.ini, /etc/php/7.4/apache2/conf.d/20-exif.ini, /etc/php/7.4/apache2/conf.d/20-ffi.ini, /etc/php/7.4/apache2/conf.d/20-fileinfo.ini, /etc/php/7.4/apache2/conf.d/20-ftp.ini, /etc/php/7.4/apache2/conf.d/20-gd.ini, /etc/php/7.4/apache2/conf.d/20-gettext.ini, /etc/php/7.4/apache2/conf.d/20-gmp.ini, /etc/php/7.4/apache2/conf.d/20-iconv.ini, /etc/php/7.4/apache2/conf.d/20-igbinary.ini, /etc/php/7.4/apache2/conf.d/20-imagick.ini, /etc/php/7.4/apache2/conf.d/20-intl.ini, /etc/php/7.4/apache2/conf.d/20-json.ini, /etc/php/7.4/apache2/conf.d/20-mbstring.ini, /etc/php/7.4/apache2/conf.d/20-msgpack.ini, /etc/php/7.4/apache2/conf.d/20-mysqli.ini, /etc/php/7.4/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.4/apache2/conf.d/20-phar.ini, /etc/php/7.4/apache2/conf.d/20-posix.ini, /etc/php/7.4/apache2/conf.d/20-readline.ini, /etc/php/7.4/apache2/conf.d/20-shmop.ini, /etc/php/7.4/apache2/conf.d/20-simplexml.ini, /etc/php/7.4/apache2/conf.d/20-soap.ini, /etc/php/7.4/apache2/conf.d/20-sockets.ini, /etc/php/7.4/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.4/apache2/conf.d/20-sysvsem.ini, /etc/php/7.4/apache2/conf.d/20-sysvshm.ini, /etc/php/7.4/apache2/conf.d/20-tokenizer.ini, /etc/php/7.4/apache2/conf.d/20-xmlreader.ini, /etc/php/7.4/apache2/conf.d/20-xmlrpc.ini, /etc/php/7.4/apache2/conf.d/20-xmlwriter.ini, /etc/php/7.4/apache2/conf.d/20-xsl.ini, /etc/php/7.4/apache2/conf.d/20-zip.ini, /etc/php/7.4/apache2/conf.d/25-memcached.ini

Installing Prestashop System Checker

Before installing Prestashop, let's install the Prestashop Checker on your current server. This is the PHP script that checks your server environment for the Prestashop installation.

Run the following command to download the Prestashop Checker script to the directory '*/var/www/html*'.

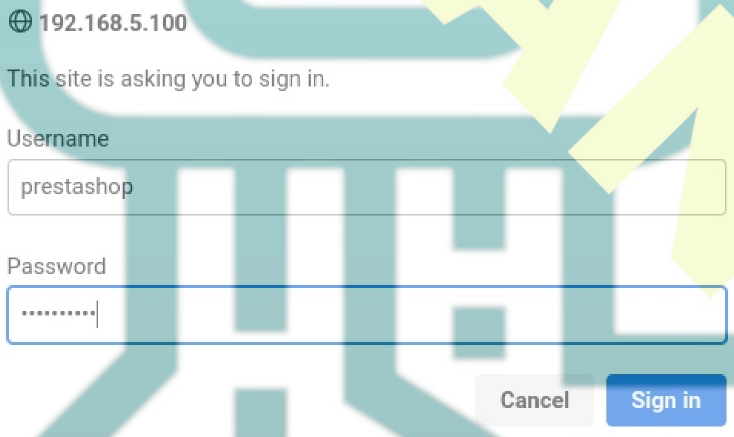
```
cd /var/www/html/
wget https://github.com/PrestaShop/php-ps-info/archive/refs/tags/v1.1.tar.gz
```

Extract the Prestashop Checker source and rename the directory to '**check-ps**'.

```
tar -xzvf v1.1.tar.gz
mv php-ps-info-1.1 check-ps
```

Back to the web browser and visit your server IP address followed by the URL path 'check-ps' (i.e: http://192.168.5.100/check-ps/phppsinfo.php).

Log in with the default user and password '**prestashop**'.


192.168.5.100

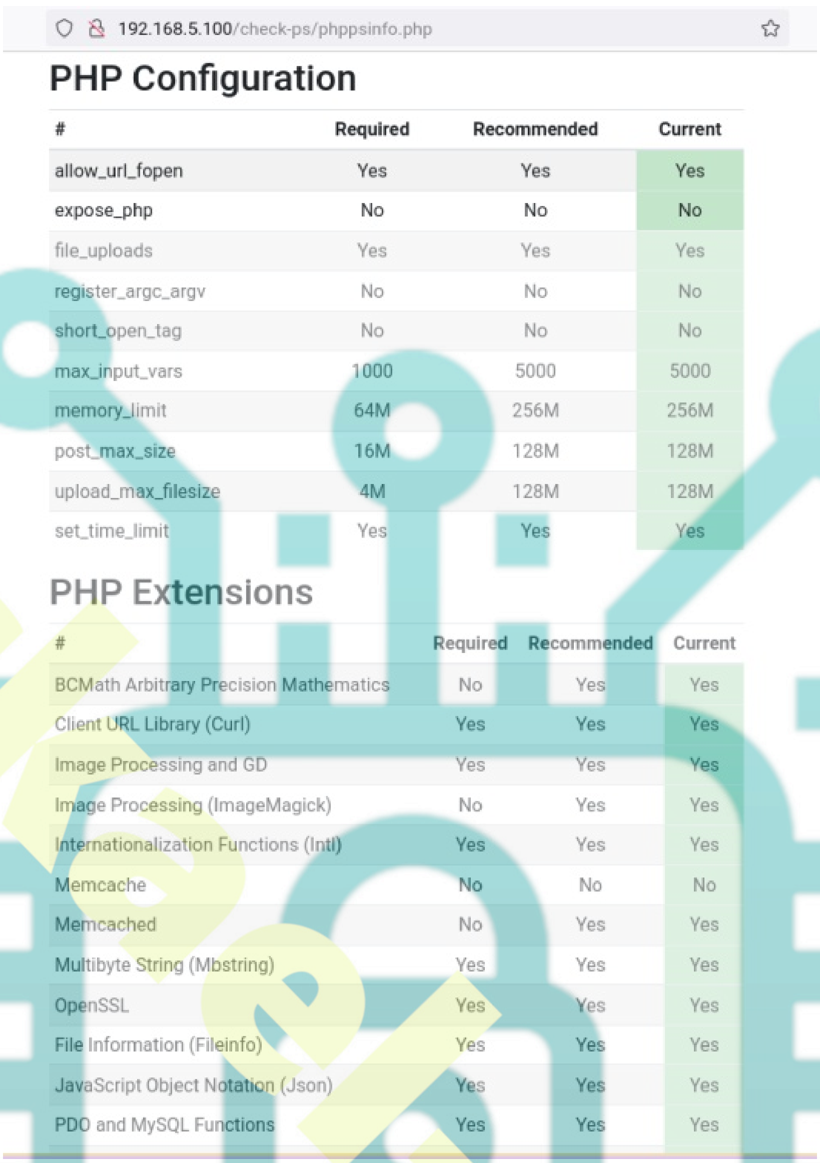
This site is asking you to sign in.

Username: prestashop

Password: ••••••••••

Now ensure all of your LAMP Stack configurations are met with the Prestashop requirements.

If some settings missing, you can edit the PHP config file '*/etc/php/8.1/apache2/php.ini*'. Also, you can install PHP extensions if there is an extension missing.


192.168.5.100/check-ps/phppsinfo.php

PHP Configuration

#	Required	Recommended	Current
allow_url_fopen	Yes	Yes	Yes
expose_php	No	No	No
file_uploads	Yes	Yes	Yes
register_argc_argv	No	No	No
short_open_tag	No	No	No
max_input_vars	1000	5000	5000
memory_limit	64M	256M	256M
post_max_size	16M	128M	128M
upload_max_filesize	4M	128M	128M
set_time_limit	Yes	Yes	Yes

PHP Extensions

#	Required	Recommended	Current
BCMath Arbitrary Precision Mathematics	No	Yes	Yes
Client URL Library (Curl)	Yes	Yes	Yes
Image Processing and GD	Yes	Yes	Yes
Image Processing (ImageMagick)	No	Yes	Yes
Internationalization Functions (Intl)	Yes	Yes	Yes
Memcache	No	No	No
Memcached	No	Yes	Yes
Multibyte String (Mbstring)	Yes	Yes	Yes
OpenSSL	Yes	Yes	Yes
File Information (Fileinfo)	Yes	Yes	Yes
JavaScript Object Notation (Json)	Yes	Yes	Yes
PDO and MySQL Functions	Yes	Yes	Yes

Creating MariaDB Database and User

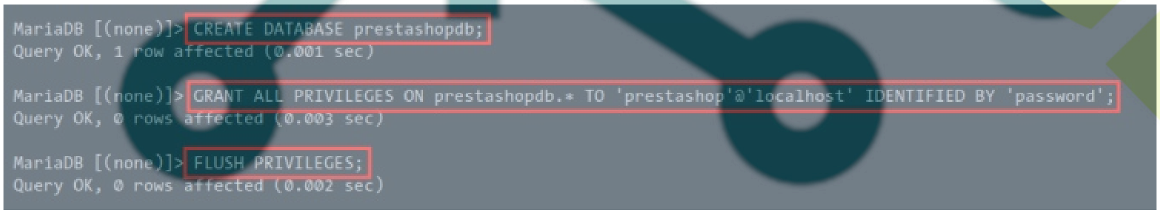
Before you start installing Prestashop, you will need to create a new MariaDB database and user.

Run the following 'mysql' command to log in to the MariaDB shell as the MariaDB '**root**' user. When prompted for a password, input your password or you can just press ENTER.

```
sudo mysql -u root -p
```

Next, run the following queries to create a new database and user for Prestashop. In this example, the database for Prestashop will be '**prestashopdb**' with the MariaDB user '**prestashop**'. You can change the password here with the strong password.

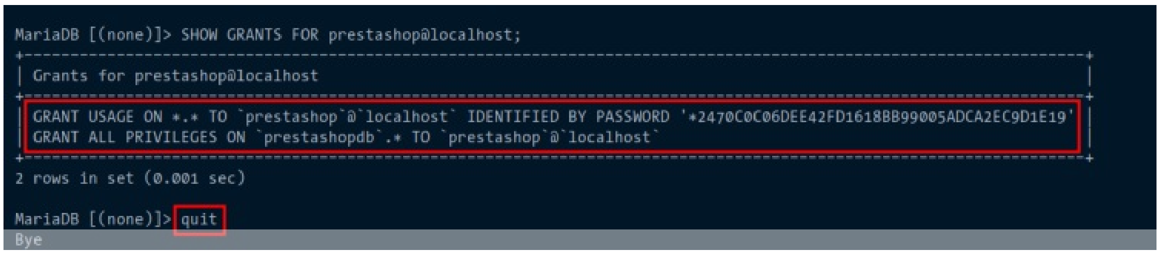
```
CREATE DATABASE prestashopdb;
GRANT ALL PRIVILEGES ON prestashopdb.* TO 'prestashop'@'localhost' IDENTIFIED BY 'password';
FLUSH PRIVILEGES;
```



```
MariaDB [(none)]> CREATE DATABASE prestashopdb;
Query OK, 1 row affected (0.001 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON prestashopdb.* TO 'prestashop'@'localhost' IDENTIFIED BY 'password';
Query OK, 0 rows affected (0.003 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.002 sec)
```

Now run the following queries to verify the privileges for the MariaDB user '**prestashop@localhost**'. Then, log out from the MariaDB shell. You should receive information about the '**prestasho@localhost**' user having privileges for the database '**prestashopdb**'.

```
SHOW GRANTS FOR prestashop@localhost;
quit
```



```
MariaDB [(none)]> SHOW GRANTS FOR prestashop@localhost;
+--------------------------------------------------------------------------------------------------------+
| Grants for prestashop@localhost                                                                        |
+--------------------------------------------------------------------------------------------------------+
| GRANT USAGE ON *.* TO `prestashop`@`localhost` IDENTIFIED BY PASSWORD '*2470C0C06DEE42FD1618BB99005ADCA2EC9D1E19' |
| GRANT ALL PRIVILEGES ON `prestashopdb`.* TO `prestashop`@`localhost`                                  |
+--------------------------------------------------------------------------------------------------------+
2 rows in set (0.001 sec)

MariaDB [(none)]> quit
Bye
```

Downloading Prestashop Source Code

After created the MariaDB database and user, you will download the Prestashop source code and setup the correct permission and ownership of the prestashp installation directory.

Now run the apt command below to install the '**unzip**' package. This will be used to extract the Prestashop source code.

```
sudo apt install unzip -y
```

Move the working directory to '*/var/www/*' and download the Prestashop source code using the wget command below.

```
cd /var/www/
wget https://download.prestashop.com/download/releases/prestashop_1.7.8.7.zip
```

Extract the Prestashop source code to the new directory '*/var/www/prestashop*' via the '*unzip*' command below.

```
unzip prestashop_1.7.8.7.zip -d /var/www/prestashop
```

Now change the ownership of the Prestashop installation directory '*/var/www/prestashop*' and change its permission. The ownership should be the user '**www-data**'and permission is '**u+rw**' (owner/user can write to the directory).

```
chown -R www-data:www-data /var/www/prestashop
chmod u+rw /var/www/prestashop
```

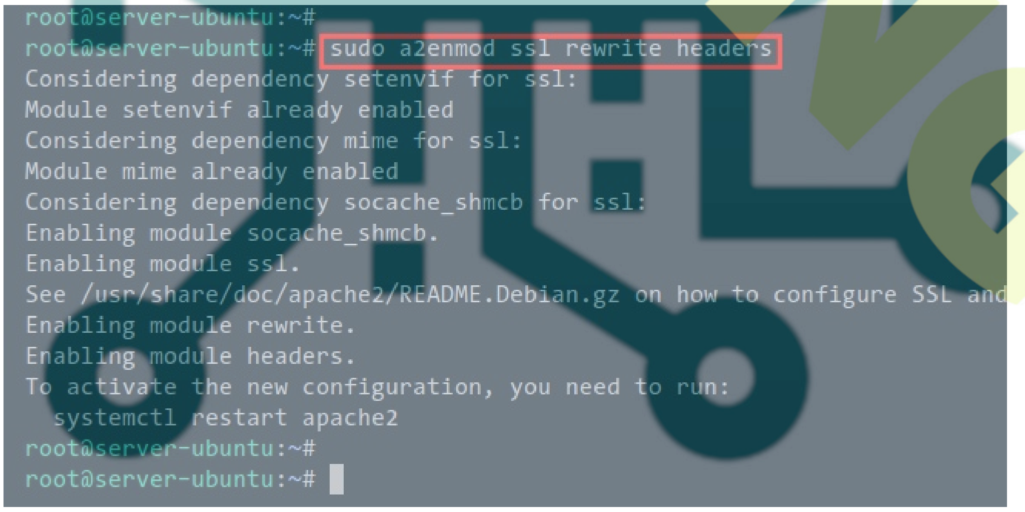
Setting Up Apache Virtual Host

All of your dependencies and configurations are ready, and the Prestashop source code is downloaded. Now you will set up the Apache virtual host for your Prestashop installation.

Before you start, ensure you have the domain name pointed to your Ubuntu server IP address and the SSL certificates generated. Also, you must enable some of the Apache2 modules for your Prestashop.

Use the following command to enable Aapche2 modules.

```
sudo a2enmod ssl rewrite headers
```



```
root@server-ubuntu:~#
root@server-ubuntu:~# sudo a2enmod ssl rewrite headers
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and
Enabling module rewrite.
Enabling module headers.
To activate the new configuration, you need to run:
  systemctl restart apache2
root@server-ubuntu:~#
root@server-ubuntu:~#
```

Create a new virtual host configuration for Prestashop '*prestashop.conf*' using the following command. All Apache2 virtual host files must be located at the '*/etc/apache2/sites-available/*' directory.

```
sudo nano /etc/apache2/sites-available/prestashop.conf
```

Add the following virtual host configurations for the prestashop. Be sure to change the domain name and the path of

SSL certificates.

```
<VirtualHost *:80>
    ServerName hwdomain.io
    Redirect permanent / https://hwdomain.io/
</VirtualHost>

<VirtualHost *:443>
    ServerAdmin admin@hwdomain.io
    DocumentRoot /var/www/prestashop
    ServerName hwdomain.io

    Protocols h2 http/1.1

    SSLEngine On
    SSLCertificateFile /etc/letsencrypt/live/hwdomain.io/fullchain.pem
    SSLCertificateKeyFile /etc/letsencrypt/live/hwdomain.io/privkey.pem

    <Directory /var/www/prestashop>
        Options +FollowSymlinks
        AllowOverride All
        Require all granted
    </Directory>

     ErrorLog /var/log/apache2/prestashop_error.log
     CustomLog /var/log/apache2/prestashop_access.log combined
</VirtualHost>
```

Save the configuration file when you are finished and close the file.

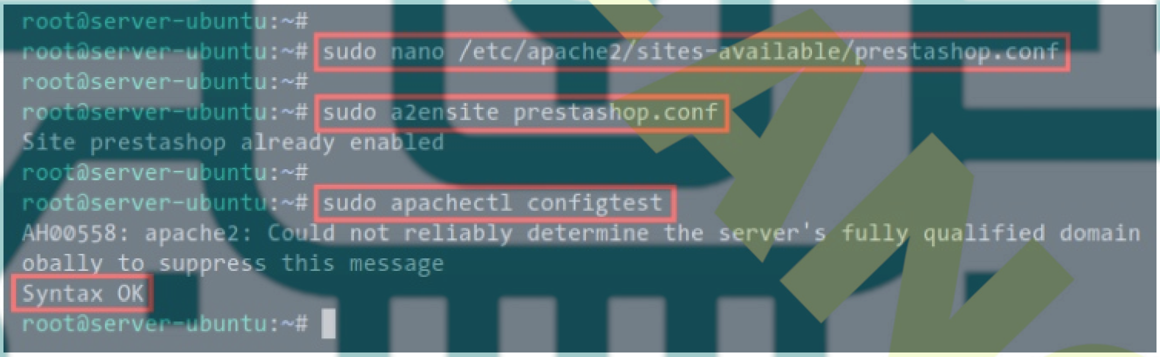
Next, run the following command to enable the virtual host file '*prestashop.conf*'. Then, verify your configurations.

You should receive an output message such as "**Syntax OK**" when your configurations have no error.

```
sudo a2ensite prestashop.conf
sudo apachectl configtest
```

Now run the following systemctl command to restart the '**apache2**' service and apply new changes. And you can start the installation wizard for Prestashop via the web browser.

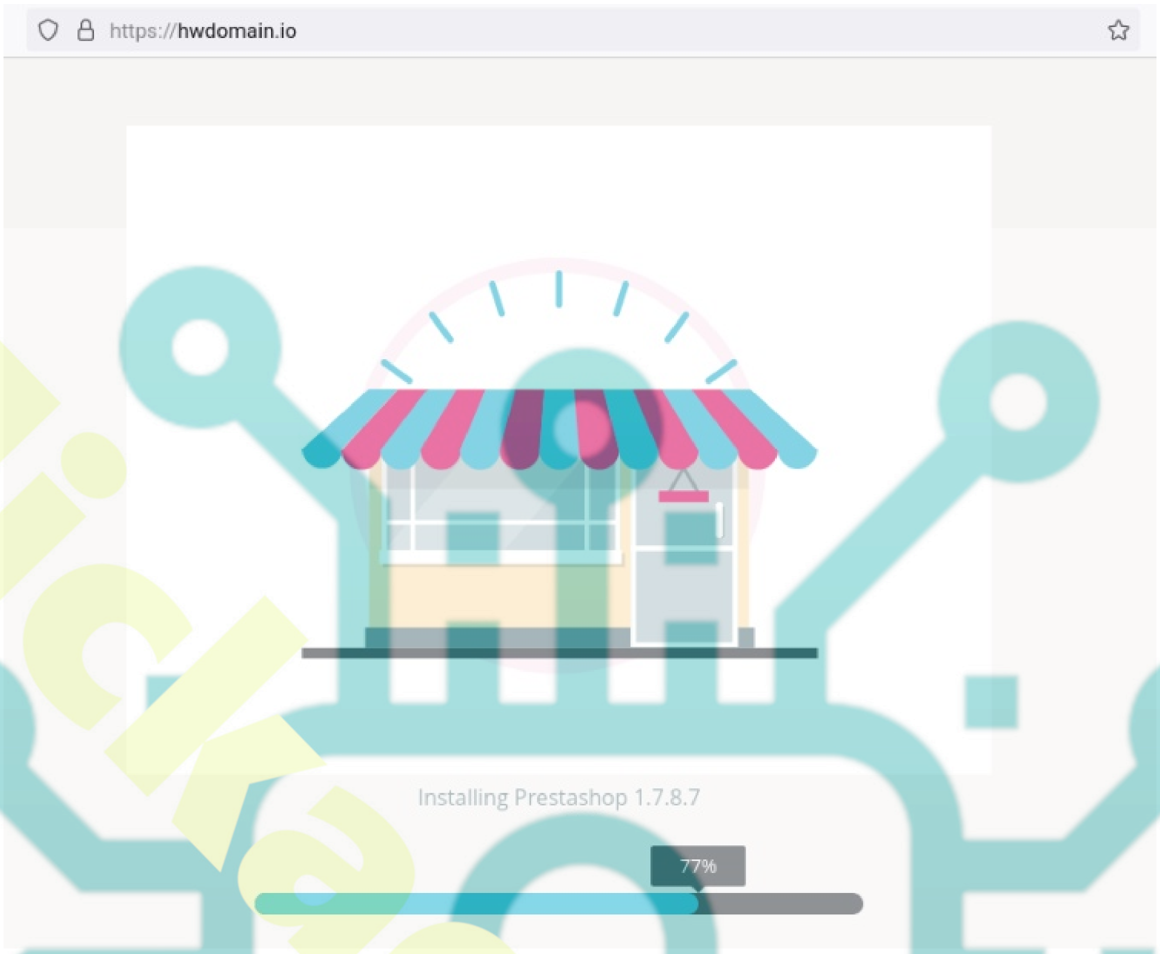
```
sudo systemctl restart apache2
```



```
root@server-ubuntu:~#
root@server-ubuntu:~# sudo nano /etc/apache2/sites-available/prestashop.conf
root@server-ubuntu:~#
root@server-ubuntu:~# sudo a2ensite prestashop.conf
Site prestashop already enabled
root@server-ubuntu:~#
root@server-ubuntu:~# sudo apachectl configtest
AH00558: apache2: Could not reliably determine the server's fully qualified domain
obally to suppress this message
Syntax OK
root@server-ubuntu:~#
```

Installing Prestashop via Web Installer

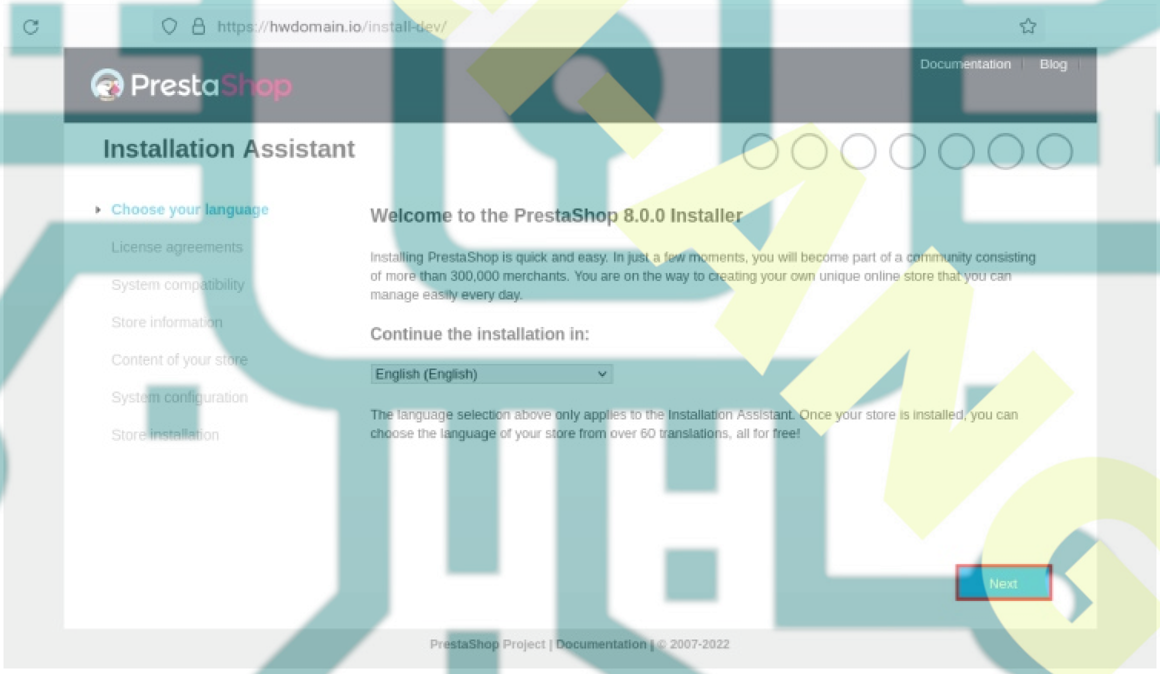
Back to your web browser and browse the domain name of your Prestashop installation (i.e: https://hwdomain.io/). You should see the web installer is starting the Prestashop installation.


https://hwdomain.io

Installing Prestashop 1.7.8.7

77%

Select the default language for your PrestaShop installation and click **Next**. The default installation language for PrestaShop is '**English**'.


https://hwdomain.io/install-dev/

Installation Assistant

Welcome to the PrestaShop 8.0.0 Installer

Installing PrestaShop is quick and easy. In just a few moments, you will become part of a community consisting of more than 300,000 merchants. You are on the way to creating your own unique online store that you can manage easily every day.

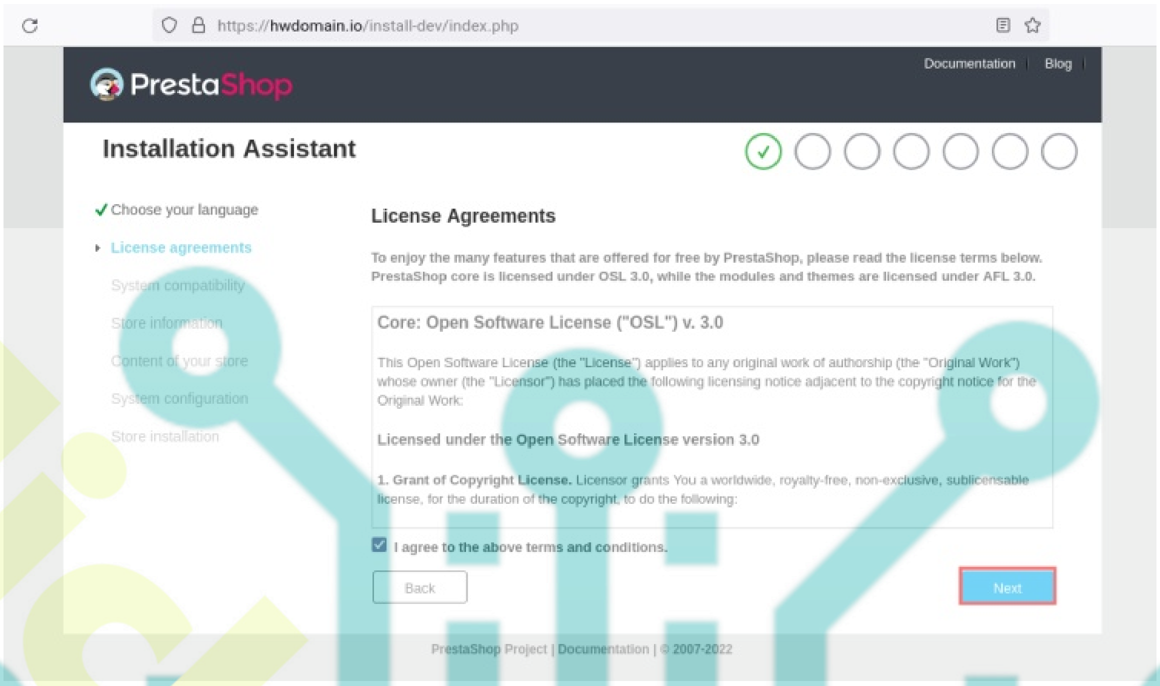
Continue the installation in:

English (English)

The language selection above only applies to the Installation Assistant. Once your store is installed, you can choose the language of your store from over 60 translations, all for free!

PrestaShop Project | Documentation | © 2007-2022

Click **agree** when showed the Prestashop License agreement page.



https://hwdomain.io/install-dev/index.php

Documentation | Blog

Installation Assistant

- ✓ Choose your language
- License agreements
- System compatibility
- Store information
- Content of your store
- System configuration
- Store installation

License Agreements

To enjoy the many features that are offered for free by PrestaShop, please read the license terms below. PrestaShop core is licensed under OSL 3.0, while the modules and themes are licensed under AFL 3.0.

Core: Open Software License ("OSL") v. 3.0

This Open Software License (the "License") applies to any original work of authorship (the "Original Work") whose owner (the "Licensor") has placed the following licensing notice adjacent to the copyright notice for the Original Work:

Licensed under the Open Software License version 3.0

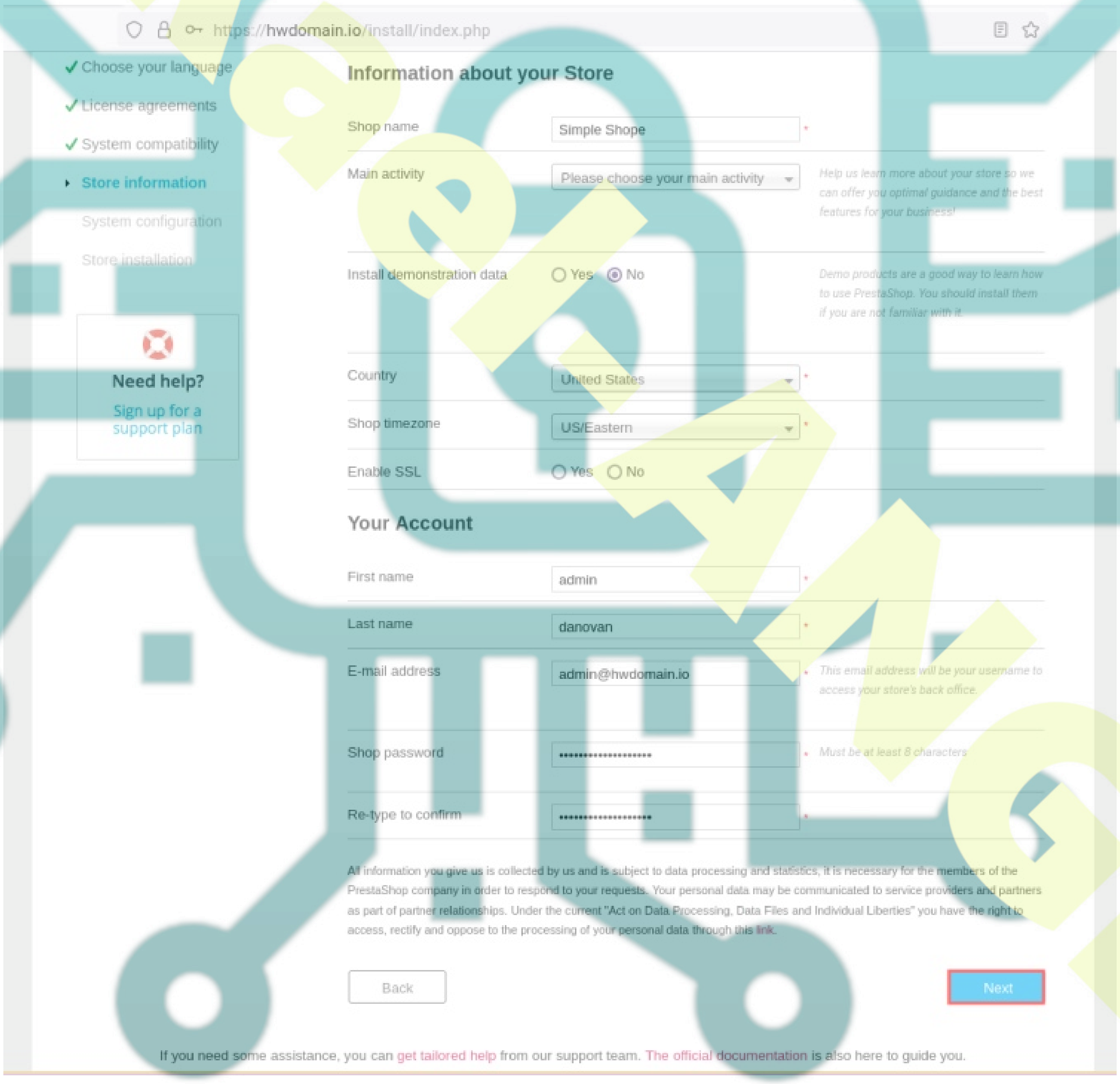
1. Grant of Copyright License. Licensor grants You a worldwide, royalty-free, non-exclusive, sublicensable license, for the duration of the copyright, to do the following:

[✓] I agree to the above terms and conditions.

Back | Next

PrestaShop Project | Documentation | © 2007-2022

Now create a new admin user for your Prestashop installation. Input details admin user, email, and password. Then, click **Next** to continue to the database configurations.



https://hwdomain.io/install/index.php

- ✓ Choose your language
- ✓ License agreements
- ✓ System compatibility
- Store information
- System configuration
- Store installation

Need help?
Sign up for a support plan

Information about your Store

Shop name	Simple Shope	*
Main activity	Please choose your main activity	Help us learn more about your store so we can offer you optimal guidance and the best features for your business!
Install demonstration data	Yes [✓ No]	Demo products are a good way to learn how to use PrestaShop. You should install them if you are not familiar with it.
Country	United States	*
Shop timezone	US/Eastern	*
Enable SSL	Yes No	

Your Account

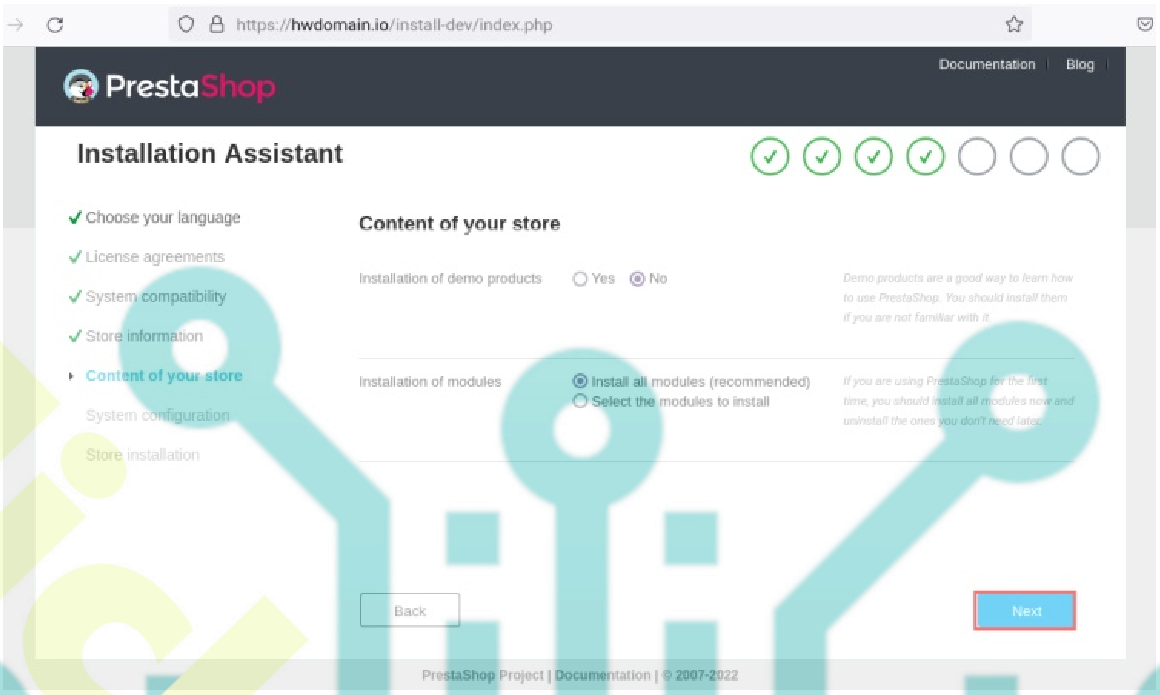
First name	admin	*
Last name	danovan	*
E-mail address	admin@hwdomain.io	* This email address will be your username to access your store's back office.
Shop password	••••••••••••••••••••	* Must be at least 8 characters
Re-type to confirm	••••••••••••••••••••	*

All information you give us is collected by us and is subject to data processing and statistics, it is necessary for the members of the PrestaShop company in order to respond to your requests. Your personal data may be communicated to service providers and partners as part of partner relationships. Under the current "Act on Data Processing, Data Files and Individual Liberties" you have the right to access, rectify and oppose to the processing of your personal data through this link.

Back | Next

If you need some assistance, you can get tailored help from our support team. The official documentation is also here to guide you.

For the store content, you can select '**No**" and click '**Next**'.


https://hwdomain.io/install-dev/index.php

Documentation | Blog

Installation Assistant

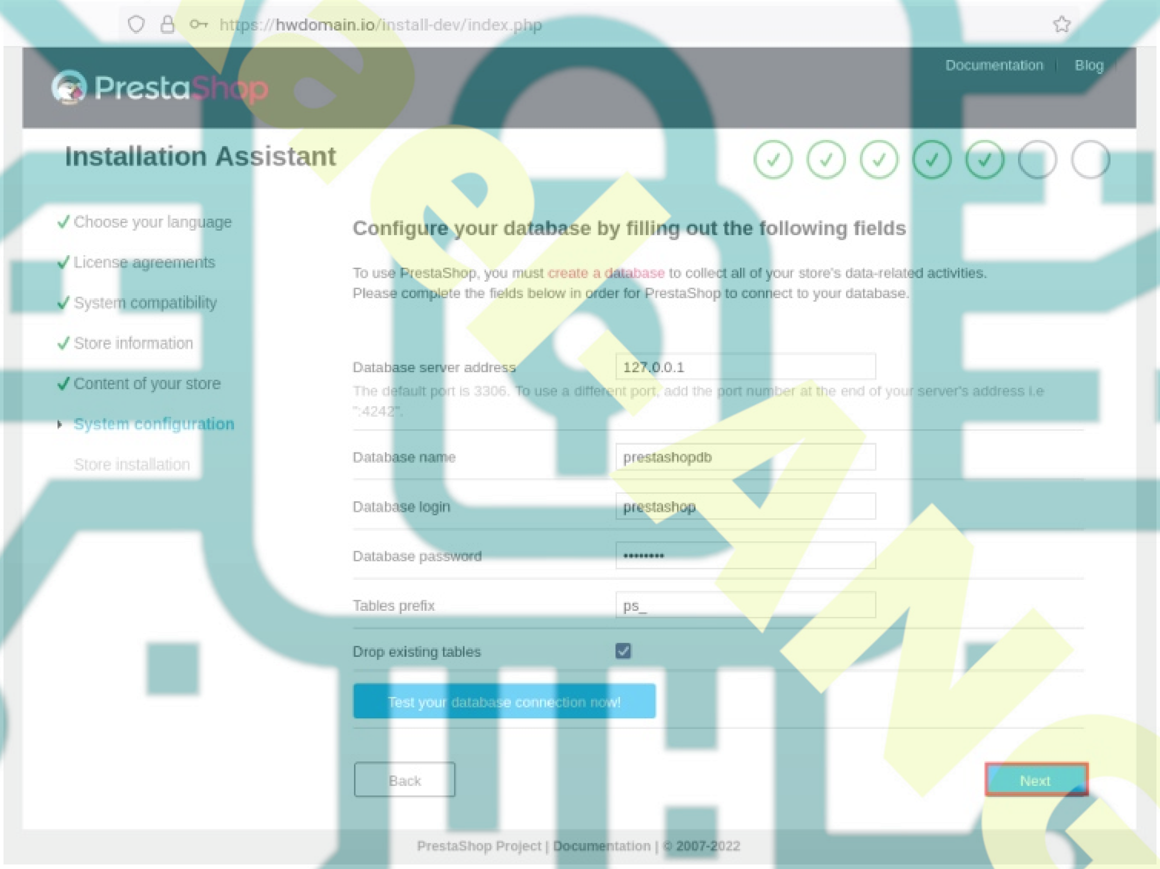
- ✓ Choose your language
- ✓ License agreements
- ✓ System compatibility
- ✓ Store information
- Content of your store
- System configuration
- Store installation

Content of your store

Installation of demo products	Yes [✓ No]	*Demo products are a good way to learn how to use PrestaShop. You should install them if you are not familiar with it.*
Installation of modules	[✓ Install all modules (recommended)] Select the modules to install	*If you are using PrestaShop for the first time, you should install all modules now and uninstall the ones you don't need later.*

PrestaShop Project | Documentation | © 2007-2022

For the Prestashop database configuration. Input details database user, database name, and the password that you have created. Then, click **Next** and the Prestashop installation will be starting.


https://hwdomain.io/install-dev/index.php

Documentation | Blog

Installation Assistant

- ✓ Choose your language
- ✓ License agreements
- ✓ System compatibility
- ✓ Store information
- ✓ Content of your store
- System configuration
- Store installation

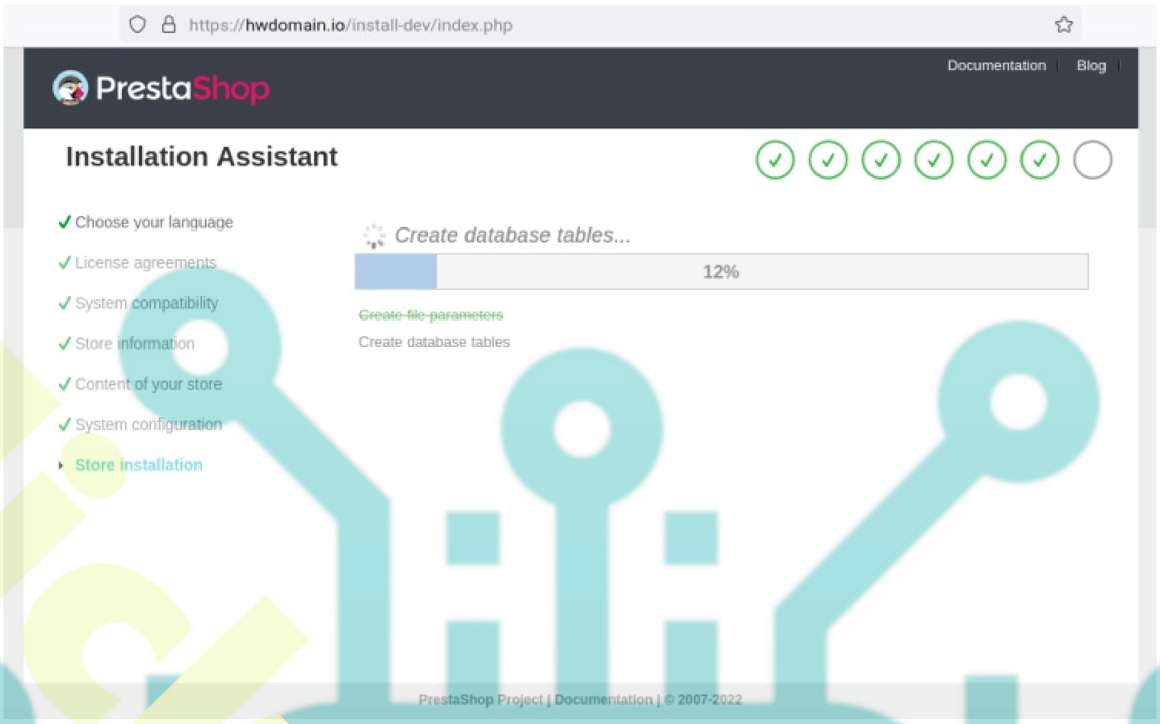
Configure your database by filling out the following fields

To use PrestaShop, you must create a database to collect all of your store's data-related activities.
Please complete the fields below in order for PrestaShop to connect to your database.

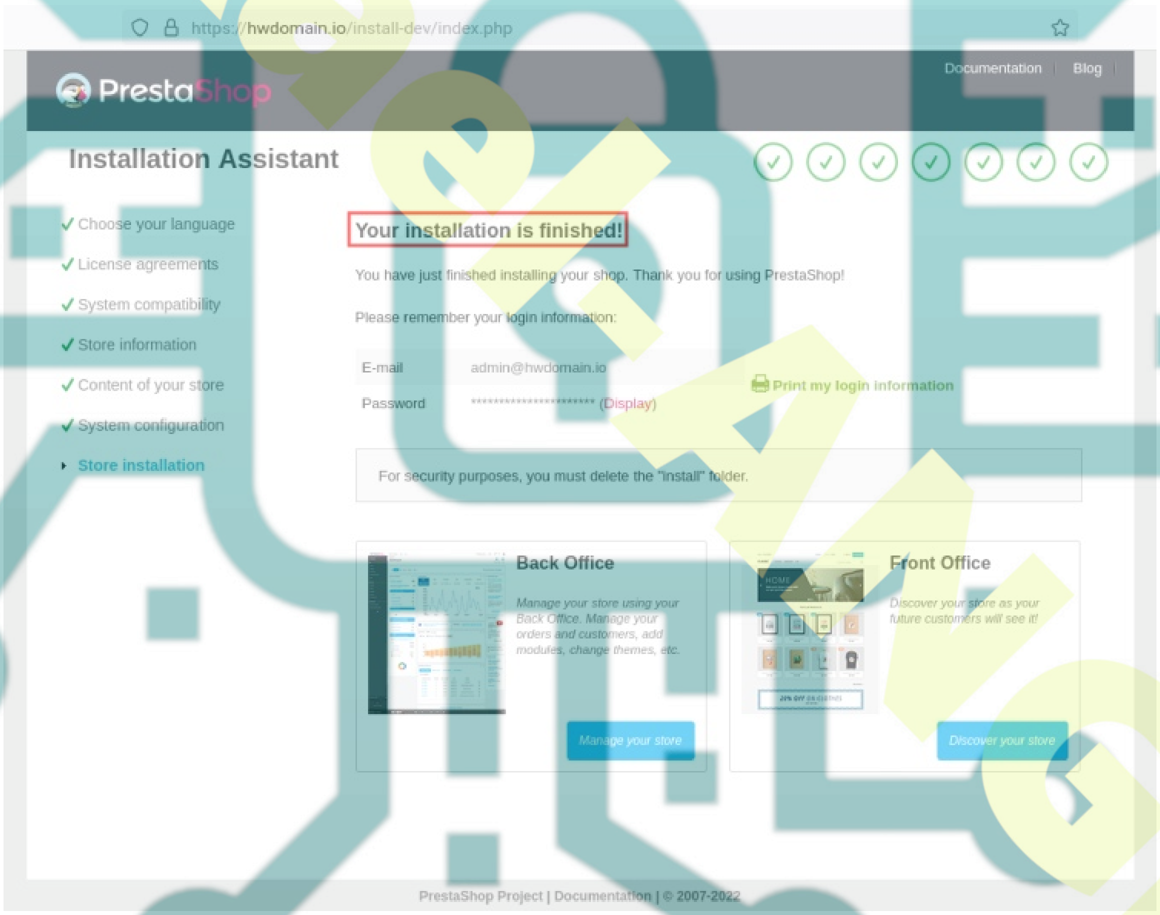
Database server address	127.0.0.1
	The default port is 3306. To use a different port, add the port number at the end of your server's address i.e ":4242".
Database name	prestashopdb
Database login	prestashop
Database password	••••••••
Tables prefix	ps_
Drop existing tables	✓

PrestaShop Project | Documentation | © 2007-2022

Below you can see the installation is processed.



When the Prestashop installation is finished, you will see the page that informs you about your Prestashop installation. Also, there is an additional change that you should do, which is deleting the '**install**' directory on your Prestashop document root.



Your installation is finished!

You have just finished installing your shop. Thank you for using PrestaShop!

Please remember your login information:

E-mail	admin@hwdomain.io
Password	********************** (Display)

For security purposes, you must delete the "install" folder.

Manage your store using your Back Office. Manage your orders and customers, add modules, change themes, etc.

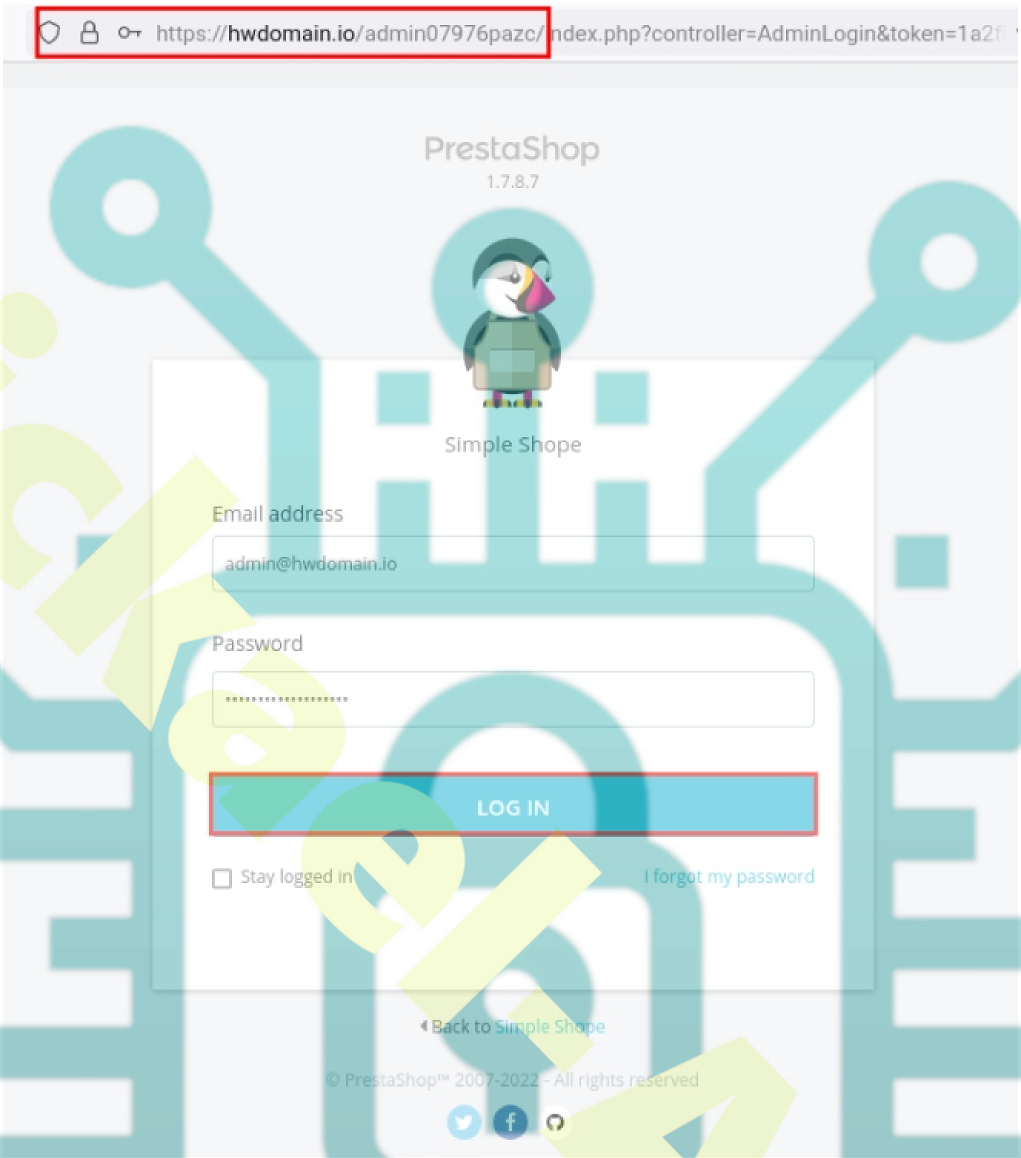
Discover your store as your future customers will see it!

Back to your terminal server and run the following command to make some of the Prestashop directory writable and delete the Prestashop '**install**' directory.

```
chmod u+rw /var/www/prestashop/var/cache
chmod u+rw /var/www/prestashop/var/logs
chmod u+rw /var/www/prestashop/img
chmod u+rw /var/www/prestashop/mails
chmod u+rw /var/www/prestashop/modules
chmod u+rw /var/www/prestashop/translations
chmod u+rw /var/www/prestashop/upload
chmod u+rw /var/www/prestashop/download
chmod u+rw /var/www/prestashop/app/config
chmod u+rw /var/www/prestashop/app/Resources/translations

rm -rf /var/www/prestashop/install
```

Back to the web browser and click the button '**Manage your store**' to access the Prestashop administration login. You will be redirected to the new Prestashop login page, and be sure to copy the path of your Prestashop admin URL.

Input the admin user and password for your Prestashop and click '**LOG IN**'.



https://hwdomain.io/admin07976pazc/index.php?controller=AdminLogin&token=1a2f

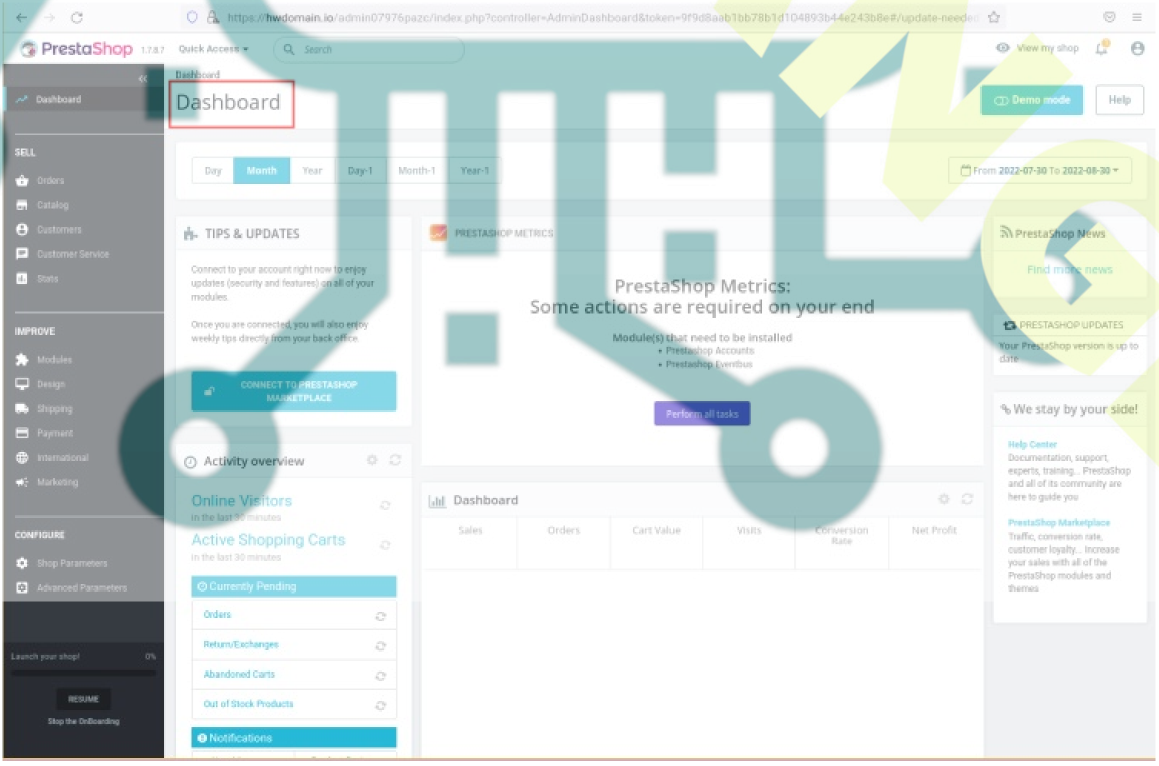
PrestaShop
1.7.8.7

Simple Shope

Email address: admin@hwdomain.io

Password: ••••••••••••••••••••

You will get the administration dashboard of Prestashop.



https://hwdomain.io/admin07976pazc/index.php?controller=AdminDashboard&token=9f9d8aab1bb78b1d104893b44e243b8e#/update-needed

From 2022-07-30 To 2022-08-30

TIPS & UPDATES

Connect to your account right now to enjoy updates (security and features) on all of your modules.

Once you are connected, you will also enjoy weekly tips directly from your back office.

PrestaShop Metrics:
Some actions are required on your end

Module(s) that need to be installed
- Prestashop Accounts
- Prestashop Eventbus

Your PrestaShop version is up to date

We stay by your side!

Help Center
Documentation, support, experts, training... PrestaShop and all of its community are here to guide you

PrestaShop Marketplace
Traffic, conversion rate, customer loyalty... Increase your sales with all of the PrestaShop modules and themes

Activity overview

Online Visitors
in the last 30 minutes

Active Shopping Carts
in the last 30 minutes

Dashboard

Sales	Orders	Cart Value	Visits	Conversion Rate	Net Profit

At this point, you have finished the installation of the open-source eCommerce solution Prestashop on Ubuntu 22.04.

Conclusion

Throughout this tutorial, you have learned how to install the eCommerce solution Prestashop on an Ubuntu 22.04 server. You have also learned the basic installation and configuration of the LAMP Stack (Apache, MariaDB, and PHP) on the Ubuntu system and learned the how to secure MariaDB server and setup Apache virtual host.

Ultimately, you have the Prestashop eCommerce solution running with LAMP Stack on an Ubuntu server and secured via SSL certificates. You can now add your themes for your customization and add plugins to extend your eCommerce website. Then, you can add your products.