

Thursday, 29 February 2024

Wordpress Hosting On EC2 Ubuntu

Launch an EC2 instance and connect to it.

```
#sudo -i
```

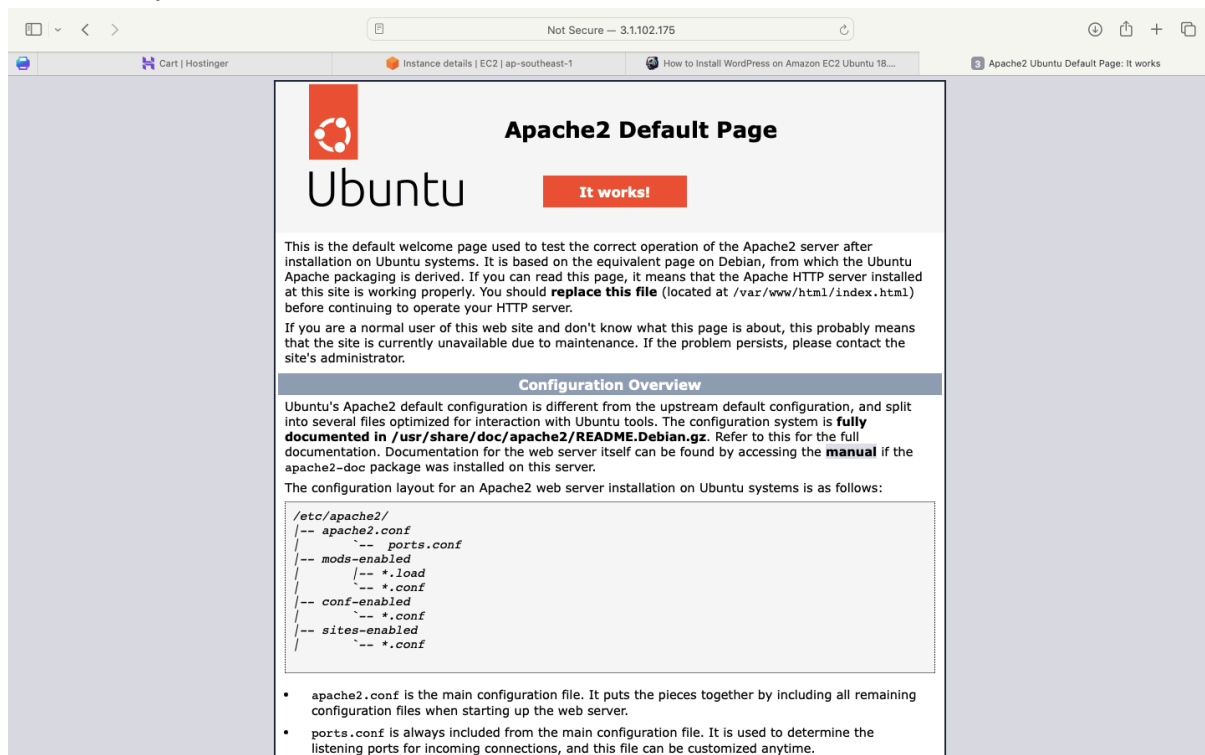
Update all packages

```
#apt update
```

Install apache

```
#apt install apache2
```

Hit PublicIp of Instance



Install php

```
#apt install php
```

Install library for php

```
#apt install libapache2-mod-php
```

Install php-mysql

```
#apt install php-mysql
```

Install mysql-server
#apt install mysql-server

Login to mysql
#mysql -u root

```
root@ip-172-31-36-32:~# mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.36-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password by 'Testpassword@123'
[ -> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password by 'Testpassword@123'
-> ^C
mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password by 'Testpassword@123';
Query OK, 0 rows affected (0.02 sec)

mysql> CREATE USER 'wp_user'@localhost IDENTIFIED BY 'Testpassword@123';
Query OK, 0 rows affected (0.03 sec)

mysql> CREATE DATABASE wp;
Query OK, 1 row affected (0.01 sec)

mysql> GRANT ALL PRIVILEGES ON wp.* TO 'wp_user'@localhost;
Query OK, 0 rows affected (0.01 sec)

mysql> ^C
mysql> ^C
mysql> ^C
mysql> exit;
Bye
```

#cd /tmp
#ls

```
~#
root@ip-172-31-36-32:~# cd /tmp
root@ip-172-31-36-32:/tmp# ls
snap-private-tmp
systemd-private-4bbb0cf465514774900009e6e4179f4a-apache2.service-ehPq9S
systemd-private-4bbb0cf465514774900009e6e4179f4a-chrony.service-diYKjH
systemd-private-4bbb0cf465514774900009e6e4179f4a-systemd-logind.service-8WZV1C
systemd-private-4bbb0cf465514774900009e6e4179f4a-systemd-resolved.service-PeSKmh
tmp.lmaPciLY81
```

Download wordpress latest package using following command.
#wget <https://wordpress.org/latest.zip>

#unzip latest.zip

```
root@ip-172-31-36-32:/tmp# unzip latest.zip
```

```

root@ip-172-31-36-32:/tmp# wget https://wordpress.org/latest.zip
--2024-02-29 04:17:47-- https://wordpress.org/latest.zip
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 25960271 (25M) [application/zip]
Saving to: 'latest.zip'

latest.zip          100%[=====>]  24.76M  6.21MB/s   in 5.5s

2024-02-29 04:17:54 (4.50 MB/s) - 'latest.zip' saved [25960271/25960271]

```

#ls

```

root@ip-172-31-36-32:/tmp# ls
latest.zip
snap-private-tmp
systemd-private-4bbb0cf465514774900009e6e4179f4a-apache2.service-ehPq9S
systemd-private-4bbb0cf465514774900009e6e4179f4a-chrony.service-diYKjH
systemd-private-4bbb0cf465514774900009e6e4179f4a-systemd-logind.service-8WZV1C
systemd-private-4bbb0cf465514774900009e6e4179f4a-systemd-resolved.service-PeSKmh
tmp.lmaPciLY81
wordpress

```

Move wordpress to /var/www/html

#mv wordpress /var/www/html

Restart apache2

#systemctl restart apache2

#cd /var/www/html

#ls

```

root@ip-172-31-36-32:/tmp# mv wordpress /var/www/html
root@ip-172-31-36-32:/tmp# systemctl restart apache2
root@ip-172-31-36-32:/tmp# cd /var/www/html
root@ip-172-31-36-32:/var/www/html# ls
index.html  wordpress

```

#cd wordpress

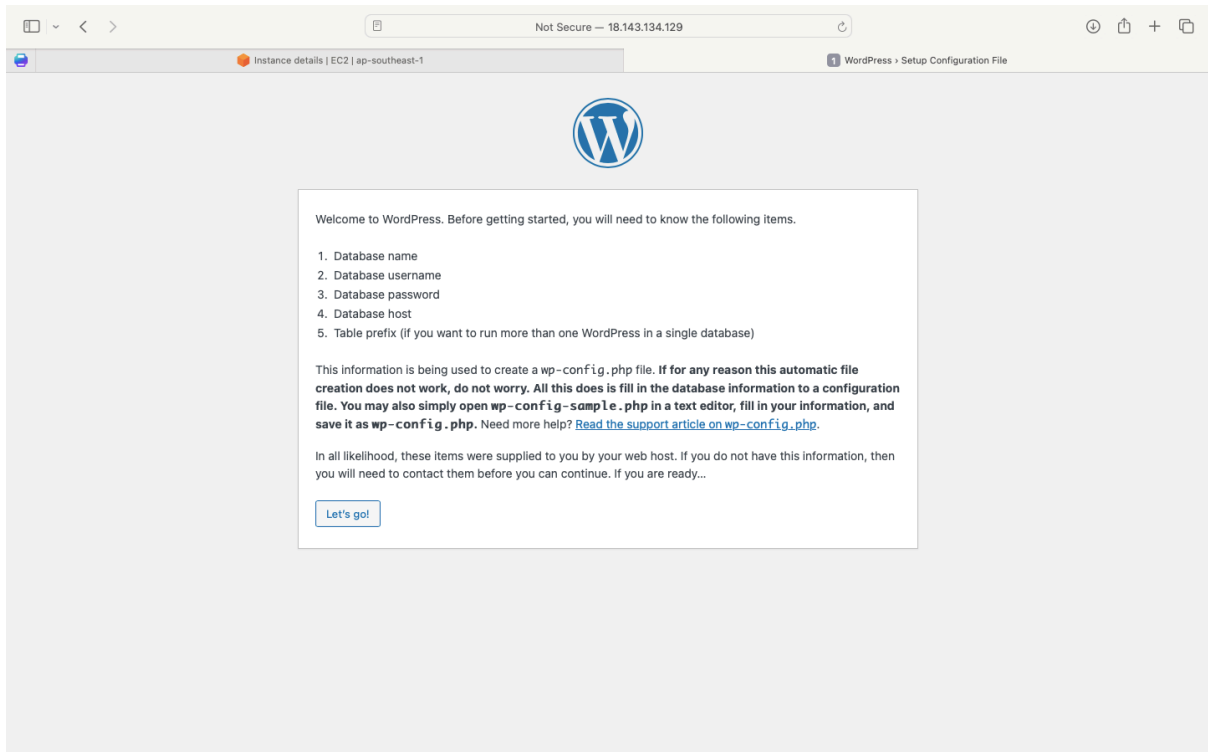
#ls

```

root@ip-172-31-36-32:/var/www/html# cd wordpress
root@ip-172-31-36-32:/var/www/html/wordpress# ls
index.php      wp-blog-header.php  wp-includes      wp-settings.php
license.txt    wp-comments-post.php wp-links-opml.php wp-signup.php
readme.html    wp-config-sample.php wp-load.php       wp-trackback.php
wp-activate.php wp-content          wp-login.php      xmlrpc.php
wp-admin       wp-cron.php         wp-mail.php

```

Hit PublicIP/wordpress



Welcome to WordPress. Before getting started, you will need to know the following items.

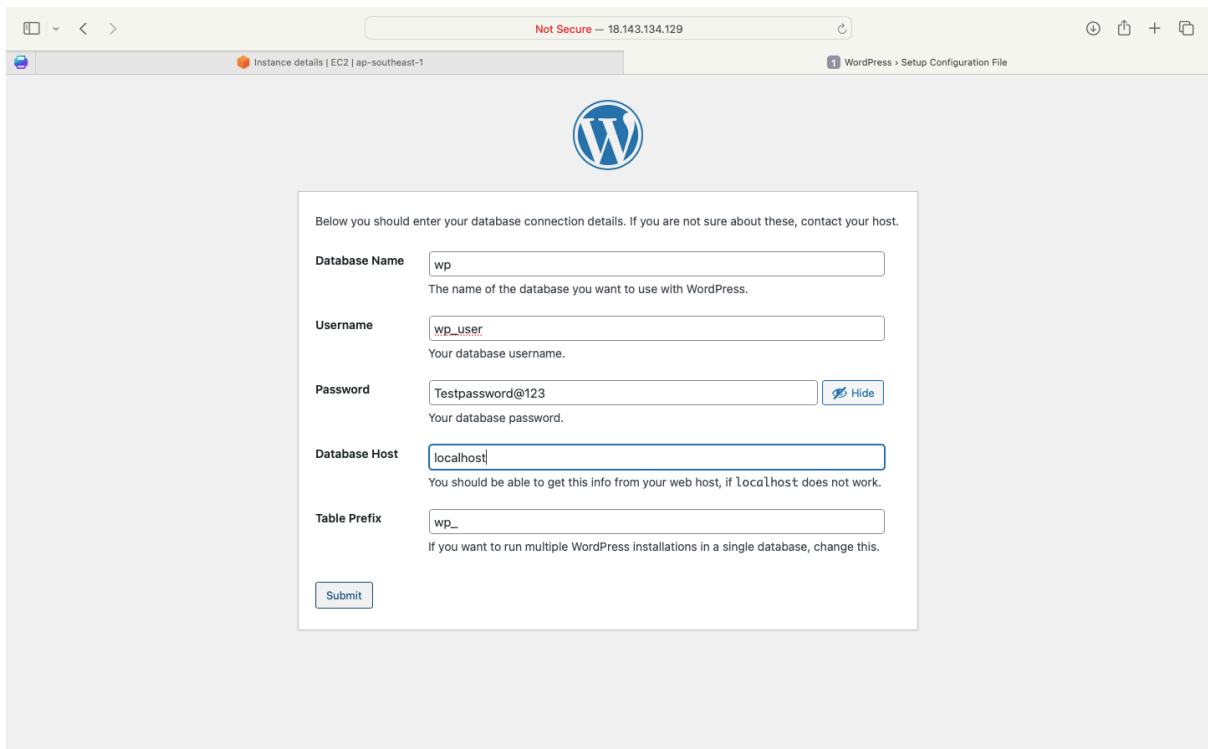
1. Database name
2. Database username
3. Database password
4. Database host
5. Table prefix (if you want to run more than one WordPress in a single database)

This information is being used to create a `wp-config.php` file. **If for any reason this automatic file creation does not work, do not worry. All this does is fill in the database information to a configuration file. You may also simply open `wp-config-sample.php` in a text editor, fill in your information, and save it as `wp-config.php`.** Need more help? [Read the support article on wp-config.php](#).

In all likelihood, these items were supplied to you by your web host. If you do not have this information, then you will need to contact them before you can continue. If you are ready...

[Let's go!](#)

Lets go



Below you should enter your database connection details. If you are not sure about these, contact your host.

Database Name
The name of the database you want to use with WordPress.

Username
Your database username.

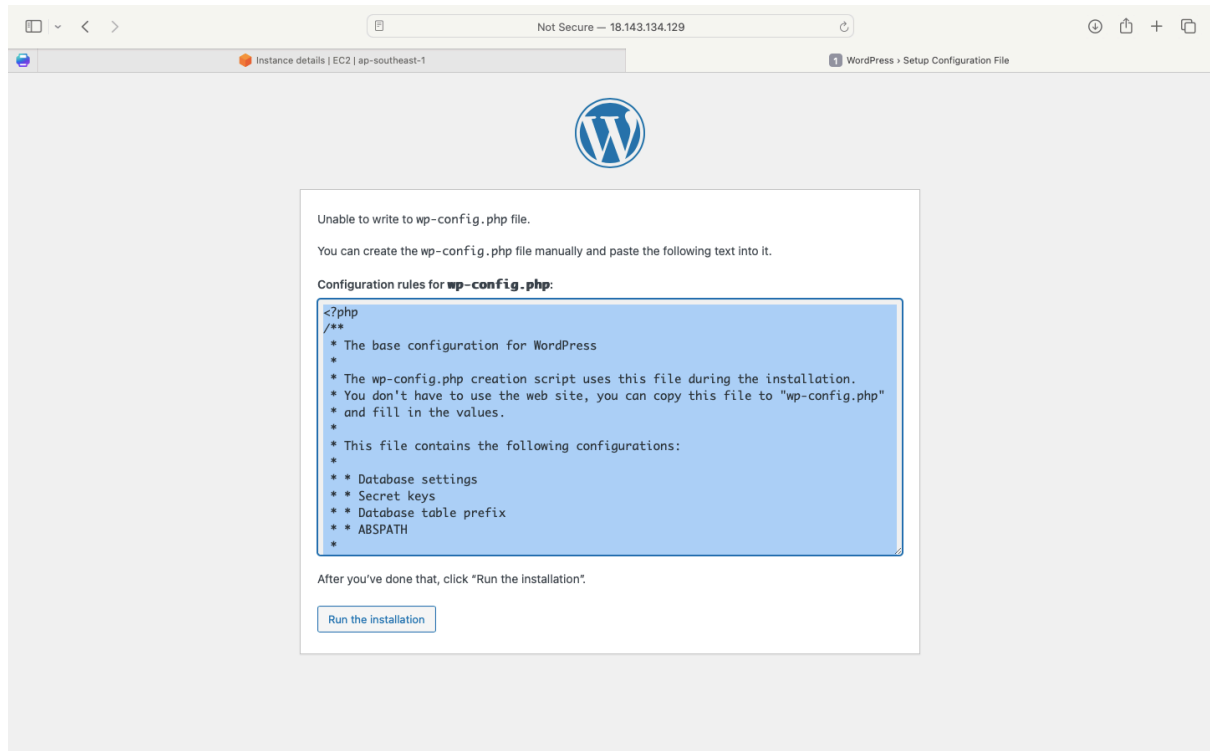
Password [Hide](#)
Your database password.

Database Host
You should be able to get this info from your web host, if `localhost` does not work.

Table Prefix
If you want to run multiple WordPress installations in a single database, change this.

[Submit](#)

Submit

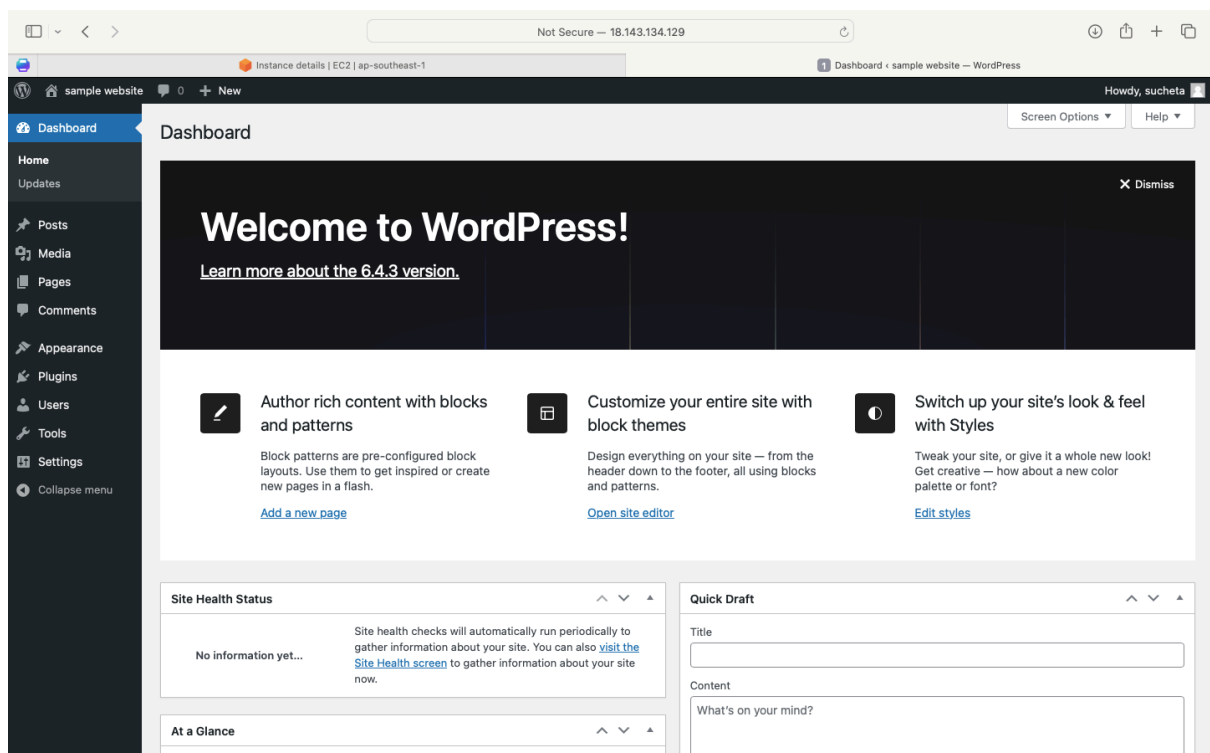


Copy this content and paste it in /var/www/html/wordpress/wp-config.php

#vim wp-config.php

#systemctl restart apache2

Again check the IP and enter login details and username of yours and submit



#etc/apache2/sites-available

```
root@ip-172-31-36-32:/etc/apache2/sites-available# tc/apache2/sites-available/
```

#vim default-ssl.conf

```
<IfModule mod_ssl.c>
    <VirtualHost _default_:443>
        ServerAdmin webmaster@localhost

        DocumentRoot /var/www/html/wordpress

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        # For most configuration files from conf-available/, which are
        # enabled or disabled at a global level, it is possible to
        # include a line for only one particular virtual host. For example the
        # following line enables the CGI configuration for this host only
        # after it has been globally disabled with "a2disconf".
        #Include conf-available/serve-cgi-bin.conf

        #
        # SSL Engine Switch:
        # Enable/Disable SSL for this virtual host.
        SSLEngine on

        #
        # A self-signed (snakeoil) certificate can be created by installing
        # the ssl-cert package. See
        # /usr/share/doc/apache2/README.Debian.gz for more info.
        # If both key and certificate are stored in the same file, only the
        # SSLCertificateFile directive is needed.
        SSLCertificateFile /etc/ssl/certs/ssl-cert-snakeoil.pem
        SSLCertificateKeyFile /etc/ssl/private/ssl-cert-snakeoil.key

        #
        # Server Certificate Chain:
        # Point SSLCertificateChainFile at a file containing the
        # concatenation of PEM encoded CA certificates which form the
```

-- INSERT --

1,21

Top

Just change Document Root from /var/www/html to /var/www/html/wordpress and save.

Hit PublicIP/wordpress

