

Main project - Using ansible

Instructions:

1. Install the necessary packages, which include MySQL 10.6, PHP version 8 or higher, and Nginx on your Linux server.
2. Create a user's home directory under the '/home' directory. You should set the root directory for the website to be '/home/username/websitename/public.'
3. Configure SFTP (Secure File Transfer Protocol) access for the user to ensure secure file management.
4. Configure phpmyadmin access for the user to ensure secure database management.
4. Install a free SSL certificate to enable secure HTTPS access to your website.
5. Once the website is hosted, proceed to the WordPress dashboard to make necessary configurations and customizations.
6. Write a personal blog post about yourself using the WordPress content management system.

Please share the following credentials after you finish the test.

Credentials:

- SFTP Credentials:
- Host:
- Username:
- Password:
- phpMyAdmin Credentials:
- URL:
- Username:
- Password:
- WordPress Credentials:
- WordPress Admin URL:
- Username:
- Password:

Please ensure that you document each step of the process and any configurations made. You may be asked to provide a brief report detailing the steps you took to complete this task.

Steps

1. Login to AWS Console and create a Linux2 instance.
2. Create inbound rule needed for the security group such as ssh 22, http 80, https 443, MySQL Aurora 3306
3. Login Instance via putty
4. Create the public and private key using # ssh-keygen
5. Place it to the ec2-user authorized keys # /home/ec2-user/.ssh/authorized_keys

```
[root@ip-172-31-7-11 ~]# cat /home/ec2-user/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCIqSk0yUrygQjFdZTnxshRZUm32TRxjS3CzIi9y1Tj9x1XWF8ffyNWQa1II+WOX2OPqG4MvUrjaJzVioPYQ6Asp9ocwltghN+YMDtGNIaPRfcJduFxmi
osdjJlmlpqCGvQ0KFnP00/91ms9svigS33Q7HI+ICi5vgobjskg0JabzYUutR12ithZJ0KyAxU32xCIpWTdS+HK6ZXjLrmS083wrBuyGBLT3Ui07/rtdUdf0i7B6ffS/FH4XDkgvjly3ULKPRQmz6QFfDlw
7kUXXku6aXv8Tp996xLc9QSrhJS89HgLRjcszTz2GWo/nizho05M660+xh/Wdx key rocketmail
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQDCyMWHiI7CLb+tf60NIT35N78QYktzaZ8McQpz7uovF440g6gbv5028Hs0fWYjPr1GOouE/963+sm7VGEEt8LnYd1b6rNaFNgtT1uA9luiUrqcn2hVwb7bV
NfuGKNLXXNDT3DnMsKom3z62TyUAVNnduSovTW3HLFgAskpC91VmL4Sh9FeqYft+/xmyhymKBYUNcg/g8CC01bcSe34nfqtLoUbxT9PqP8zFbvAeIu3+4iyUyLMvLQtHzkqpKBN+Aya8U8cCT/Qn
11Q8Ac7ojRjc2zUi0xaUe2u12S4PQjXtaqjIQa+kheVui/W5oqb56ahhDbec2t root@ip-172-31-7-11.ap-south-1.compute.internal
[root@ip-172-31-7-11 ~]#
```

6. Add public key to /etc/hosts

```
[root@ip-172-31-7-11 ~]# cat /etc/hosts
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
::1 localhost6 localhost6.localdomain6
172.31.7.11
[root@ip-172-31-7-11 ~]#
```

7. Install Epel repository# amazon-linux-extras install epel -y
8. Install Ansible # yum install -y ansible
9. Create directory ansible # mkdir ansible
10. Move to ansible directory and create a inventory file for hosts

```
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]# cd ansible
[root@ip-172-31-7-11 ansible]# cat inventory
[web]
172.31.7.11
[root@ip-172-31-7-11 ansible]#
```

11. Create ansible.cfg file for remote configuration

```
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ansible]# cat ansible.cfg
[defaults]
inventory=/root/ansible/inventory
remote_user=ec2-user
ask_pass=false
[privilege_escalation]
become=true
become_method=sudo
become_user=root
become_ask_pass=false

[root@ip-172-31-7-11 ansible]#
```

12. Create DNS A record using instance public ip

The screenshot shows the GoDaddy DNS Management interface. The left sidebar has categories: Domains, DNS, Transfers, Services, Tools, and Settings. The main table lists DNS records:

	Type	Name	Data	TTL	Delete	Edit
Portfolio	A	@	13.126.184.179	1 Hour		
DNS	NS	@	ns53.domaincontrol.com.	1 Hour	Can't delete	Can't edit
Transfers	NS	@	ns54.domaincontrol.com.	1 Hour	Can't delete	Can't edit
Services	CNAME	www	pooja.quest.	1 Hour		
Tools	CNAME	_domainconnect	_domainconnect.gd.domaincontrol.co m.	1 Hour		

13. Create playbook

```
[root@ip-172-31-7-11 ansible]# cat playbook.yml
```

```

[root@ip-172-31-1-251:~/ansible]
[root@ip-172-31-1-251 ansible]# cat playbook.yml
---
- name: Setup LAMP stack with WordPress on Amazon Linux 2
  hosts: all
  become: yes
  vars:
    domain_name: "pooja.quest"
    sftp_pwd: "pooja$1!"
    mysql_root_pwd: "pooja$1!"
    phpadmin_pwd: "pooja$1!"
    wordpress_pwd: "pooja$1!"
    user_path: "/home/pooja"
    user_website_path: "/home/pooja/mywebsite/public"
  tasks:
    - name: Install and Enable PHP 8
      command: amazon-linux-extras enable php8.0
    - name: Install Nginx, PHP8, MySQL, and Certbot
      yum:
        name:
          - nginx
          - php
          - php-mysqlnd
          - php-fpm
          - mariadb
          - mariadb-server
          - certbot
          - certbot-nginx
          - python2-pip
          - MySQL-python
        state: present
    - name: Install required Python packages (MySQL-python package provides a Python interface to MySQL databases)
      pip:
        name: MySQL-python
        state: present
        executable: /usr/bin/pip2
    - name: Start and enable Nginx, PHP and MariaDB
      systemd:
        name: "{{ item }}"
        state: started

```

```
root@ip-172-31-1-251:~/ansible
- name: Start and enable Nginx, PHP and MariaDB
  systemd:
    name: "{{ item }}"
    state: started
    enabled: yes
  loop:
    - nginx
    - mariadb
    - php-fpm
- name: Check MariaDB is running
  systemd:
    name: mariadb
    state: started
    enabled: yes
- name: Initialize MySQL data directory
  command: mysql_install_db
  args:
    creates: /var/lib/mysql/mysql
- name: Set MySQL password for user root
  mysql_user:
    login_user: root
    login_password: "{{ mysql_root_pwd }}"
    user: root
    password: "{{ mysql_root_pwd }}"
    host_all: yes
    state: present
- name: Remove anonymous MySQL users
  mysql_user:
    name: ''
    host_all: yes
    state: absent
    login_user: root
    login_password: "{{ mysql_root_pwd }}"
- name: Remove MySQL test database
  mysql_db:
    name: test
    state: absent
    login_user: root
    login_password: "{{ mysql_root_pwd }}"

```

```
root@ip-172-31-1-251:~/ansible
- name: Create MySQL Database for user root(DB Name wordpress)
  mysql_db:
    name: wordpress
    state: present
    login_user: root
    login_password: "{{ mysql_root_pwd }}"
- name: Create MySQL user for WordPress(Here user set as wordpress)
  mysql_user:
    name: wordpress
    password: "{{ wordpress_pwd }}"
    priv: "wordpress:ALL"
    state: present
    login_user: root
    login_password: "{{ mysql_root_pwd }}"
- name: Create user for SFTP (username pooja)
  user:
    name: pooja
    home: "{{ user_path }}"
    shell: /bin/bash
    create_home: yes
    password: "{{ sftp_pwd | password_hash('sha512') }}"
- name: Create Directory for website
  file:
    path: "{{ user_website_path }}"
    state: directory
    owner: pooja
    group: root
    mode: '0755'
  notify: Restart Nginx and PHP-FPM
- name: Ensure /home/pooja has execute permission for others
  file:
    path: /home/pooja
    mode: '0751'
- name: Ensure sshd_config.d directory exists
  file:
    path: /etc/ssh/sshd_config.d
    state: directory

```

```
root@ip-172-31-1-251:~/ansible
- name: Set up SFTP access
  copy:
    content: |
      Match User pooja
      ChrootDirectory {{ user_path }}
      ForceCommand internal-sftp
      AllowTcpForwarding no
      PermitTunnel no
      X11Forwarding no
  dest: /etc/ssh/sshd_config.d/sftp_pooja.conf
#
#   - name: Create SSL certification
#     command: certbot --nginx -d {{ domain_name }} --non-interactive --agree-tos -m admin@{{ domain_name }}

- name: Ensure snippets directory exists for Nginx
  file:
    path: /etc/nginx/snippets
    state: directory
    owner: root
    group: root
    mode: '0755'
- name: Ensure fastcgi-php.conf exists (It handling PHP FastCGI parameters in Nginx)
  copy:
    content: |
      fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
      include fastcgi_params;
  dest: /etc/nginx/snippets/fastcgi-php.conf
  owner: root
  group: root
  mode: '0644'
- name: Install phpMyAdmin
  get_url:
    url: https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
    dest: /tmp/phpMyAdmin.tar.gz
- name: Extract phpMyAdmin
  unarchive:
    src: /tmp/phpMyAdmin.tar.gz
    dest: /usr/share/

```

```
root@ip-172-31-1-251:~/ansible
- name: Ensure phpMyAdmin directory exists
  file:
    path: /usr/share/phpMyAdmin
    state: directory

- name: Copy phpMyAdmin files to /usr/share/phpMyAdmin
  copy:
    src: /usr/share/phpMyAdmin-5.2.1-all-languages/
    dest: /usr/share/phpMyAdmin/
    remote_src: yes
    force: yes

- name: Create phpMyAdmin config file
  copy:
    content: |
      <?php
      $cfg['blowfish_secret'] = '({ phpmyadmin_pwd })';
      $i = 0;
      $i++;
      $cfg['Servers'][$i]['auth_type'] = 'cookie';
      $cfg['Servers'][$i]['user'] = 'root';
      $cfg['Servers'][$i]['password'] = '({ mysql_root_pwd })';
      $cfg['Servers'][$i]['host'] = 'localhost';
      $cfg['Servers'][$i]['AllowNoPassword'] = false;
    dest: /usr/share/phpMyAdmin/config.inc.php
    owner: nginx
    group: nginx
    mode: '0644'

- name: Set permission for PhpMyAdmin directory
  file:
    path: /usr/share/phpMyAdmin
    state: directory
    owner: nginx
    group: nginx
    mode: '0775'

- name: Set permissions for PHP sessions directory
  file:
    path: /var/lib/php
    state: directory
    owner: nginx
    group: nginx
    mode: '0775'


```

```
root@ip-172-31-1-251:~/ansible
- name: Set permissions for PHP sessions directory
  file:
    path: /var/lib/php
    state: directory
    owner: nginx
    group: nginx
    mode: '0775'

- name: Set ownership for PHP sessions directory
  ansible.builtin.file:
    Path: /var/lib/php
    Owner: nginx
    Group: nginx
    Recurse: yes
    Become: yes

- name: Set permissions for PHP sessions directory
  file:
    Path : /var/lib/php/session
    state: directory
    owner: nginx
    group: nginx
    mode: "0775"
    recurse: yes
  become: yes

- name: Set ownership for PHP-FPM directory
  ansible.builtin.file:
    Path: /run/php-fpm
    owner: nginx
    group: nginx
    recurse: yes
  become: yes

- name: Set permissions for PHP-FPM directory
  ansible.builtin.file:
    Path: /run/php-fpm
    mode: "0755"
    recurse: yes


```

```
root@ip-172-31-1-251:~/ansible
- name: Download and extract WordPress
  unarchive:
    src: https://wordpress.org/latest.tar.gz
    dest: "{{ user_website_path }}"
    remote_src: yes
    creates: "{{ user_website_path }}/wordpress"

- name: Configure WordPress wp-config.php
  copy:
    content: |
      <?php
      define('DB_NAME', 'wordpress');
      define('DB_USER', 'root');
      define('DB_PASSWORD', '({ wordpress_pwd })');
      define('DB_HOST', 'localhost');
      define('DB_CHARSET', 'utf8');
      define('DB_COLLATE', '');
      define('AUTH_KEY', 'put your unique phrase here');
      define('SECURE_AUTH_KEY', 'put your unique phrase here');
      define('LOGGED_IN_KEY', 'put your unique phrase here');
      define('NONCE_KEY', 'put your unique phrase here');
      define('AUTH_SALT', 'put your unique phrase here');
      define('SECURE_AUTH_SALT', 'put your unique phrase here');
      define('LOGGED_IN_SALT', 'put your unique phrase here');
      define('NONCE_SALT', 'put your unique phrase here');
      $table_prefix = 'wp_';
      define('WP_DEBUG', false);
      if (!defined('ABSPATH'))
        define('ABSPATH', dirname(__FILE__) . '/');
      require_once(ABSPATH . 'wp-settings.php');
    dest: "{{ user_website_path }}/wordpress/wp-config.php"
    owner: pooya
    group: pooya
    mode: '0644'
  notify: Restart Nginx and PHP-FPM

- name: Configure Nginx for WordPress and phpmyadmin
  copy:
    content: |
      server {
        server_name {{ domain_name }};
        root {{ user_website_path }}/wordpress;

```

```

root@ip-172-31-1-251:~/ansible
index index.php index.html index.htm
# WordPress location configuration (important fastcgi_pass unix:/run/php-fpm/www.sock)
location / {
    try_files $uri $uri/ /index.php?$args;
}

location ~ \.php$ {
    include snippets/fastcgi-php.conf;
    fastcgi_pass unix:/run/php-fpm/www.sock;
}

location ~ /\.ht {
    deny all;
}

# phpMyAdmin location configuration
location /phpmyadmin {
    alias /usr/share/phpMyAdmin/;
    index index.php index.html index.htm;

    location ~ ^/phpmyadmin/(.+\.php)$ {
        alias /usr/share/phpMyAdmin/$1;
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/run/php-fpm/www.sock;
        fastcgi_param SCRIPT_FILENAME $request_filename;
        include fastcgi_params;
    }

    location ~ ^/phpmyadmin/(index\.php)?$ {
        deny all;
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/run/php-fpm/www.sock;
        fastcgi_param SCRIPT_FILENAME $request_filename;
        include fastcgi_params;
    }
}
dest: "/etc/nginx/conf.d/{{ domain_name }}.conf"
owner: root
group: root
mode: '0644'

root@ip-172-31-1-251:~/ansible
# phpMyAdmin location configuration
location /phpmyadmin {
    alias /usr/share/phpMyAdmin/;
    index index.php index.html index.htm;

    location ~ ^/phpmyadmin/(.+\.php)$ {
        alias /usr/share/phpMyAdmin/$1;
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/run/php-fpm/www.sock;
        fastcgi_param SCRIPT_FILENAME $request_filename;
        include fastcgi_params;
    }

    location ~ ^/phpmyadmin/(index\.php)?$ {
        deny all;
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/run/php-fpm/www.sock;
        fastcgi_param SCRIPT_FILENAME $request_filename;
        include fastcgi_params;
    }
}
dest: "/etc/nginx/conf.d/{{ domain_name }}.conf"
owner: root
group: root
mode: '0644'

- name: Restart Nginx and PHP-FPM
  systemd:
    name: "{{ item }}"
    state: restarted
  loop:
    - nginx
    - php-fpm
  handlers:
    - name: Restart Nginx and PHP-FPM
      name: sshd
      state: restarted
[root@ip-172-31-1-251 ansible]#

```

14. For ssh configuration PasswordAuthentication make yes # /etc/ssh/sshd_config

15. Run the playbook # ansible-playbook -I inventory playbook.yml

```

root@ip-172-31-1-251:~/ansible
[root@ip-172-31-1-251 ansible]# ansible-playbook -i inventory playbook.yml
PLAY [Setup LAMP stack with WordPress on Amazon Linux 2] ****
TASK [Gathering Facts] ****
[WARNING]: Platform linux on host 13.126.184.179 is using the discovered Python interpreter at /usr/bin/python, but future installation of another Python interpreter could change this. See https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
ok: [13.126.184.179]

TASK [Install and Enable PHP 8] ****
changed: [13.126.184.179]

TASK [Install Nginx, PHP8, MySQL, and Certbot] ****
ok: [13.126.184.179]

TASK [Install required Python packages(MySQL-python package provides a Python interface to MySQL databases)] ***
ok: [13.126.184.179]

TASK [Start and enable Nginx, PHP and MariaDB] ****
ok: [13.126.184.179]
ok: [13.126.184.179] => (item=nginx)
ok: [13.126.184.179] => (item=mariadb)
ok: [13.126.184.179] => (item=php-fpm)

TASK [Check MariaDB is running] ****
ok: [13.126.184.179]

TASK [Initialize MySQL data directory] ****
ok: [13.126.184.179]

TASK [Set MySQL password for user root] ****
[WARNING]: Module did not set no_log for update_password
ok: [13.126.184.179]

TASK [Remove anonymous MySQL users] ****
ok: [13.126.184.179]

TASK [Remove MySQL test database] ****
ok: [13.126.184.179]

TASK [Create MySQL Database for user root(DB Name wordpress)] ****
ok: [13.126.184.179]

```

```

root@ip-172-31-1-251:~/ansible
TASK [Copy phpMyAdmin files to /user/share/phpMyAdmin] ****
ok: [13.126.184.179]
TASK [Create phpMyAdmin config file] ****
ok: [13.126.184.179]
TASK [Set permission for PhpMyAdmin directory] ****
ok: [13.126.184.179]
TASK [Set permissions for PHP sessions directory] ****
ok: [13.126.184.179]
TASK [Set ownership for PHP sessions directory] ****
ok: [13.126.184.179]
TASK [Set permissions for PHP sessions directory] ****
ok: [13.126.184.179]
TASK [Set ownership for PHP-FPM directory] ****
changed: [13.126.184.179]
TASK [Set permissions for PHP-FPM directory] ****
changed: [13.126.184.179]
TASK [Download and extract WordPress] ****
skipping: [13.126.184.179]
TASK [Configure Wordpress wp-config.php] ****
changed: [13.126.184.179]
TASK [Configure Nginx for WordPress and phpmyadmin] ****
ok: [13.126.184.179]
TASK [Restart Nginx and PHP-FPM] ****
changed: [13.126.184.179] => (item=nginx)
changed: [13.126.184.179] => (item=php-fpm)
RUNNING HANDLER [Restart Nginx and PHP-FPM] ****
changed: [13.126.184.179]

PLAY RECAP ****
13.126.184.179 : ok=33    changed=7     unreachable=0    failed=0    skipped=2    rescued=0    ignored=0
[root@ip-172-31-1-251 ansible]#

```

Verify Output

1. Check the versions

```

[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]# mysql --version
mysql Ver 15.1 Distrib 5.5.68-MariaDB, for Linux (x86_64) using readline 5.1
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]# php -v
PHP 8.0.30 (cli) (built: Aug 24 2023 20:32:30) ( NTS )
Copyright (c) The PHP Group
Zend Engine v4.0.30, Copyright (c) Zend Technologies
[root@ip-172-31-7-11 ~]#

```

2. Verify users home directory and website directory if present or not

```

[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]# cd /home/pooja/mywebsite/public
[root@ip-172-31-7-11 public]# ls
wordpress
[root@ip-172-31-7-11 public]#

```

3. Verify SFTP with filezilla

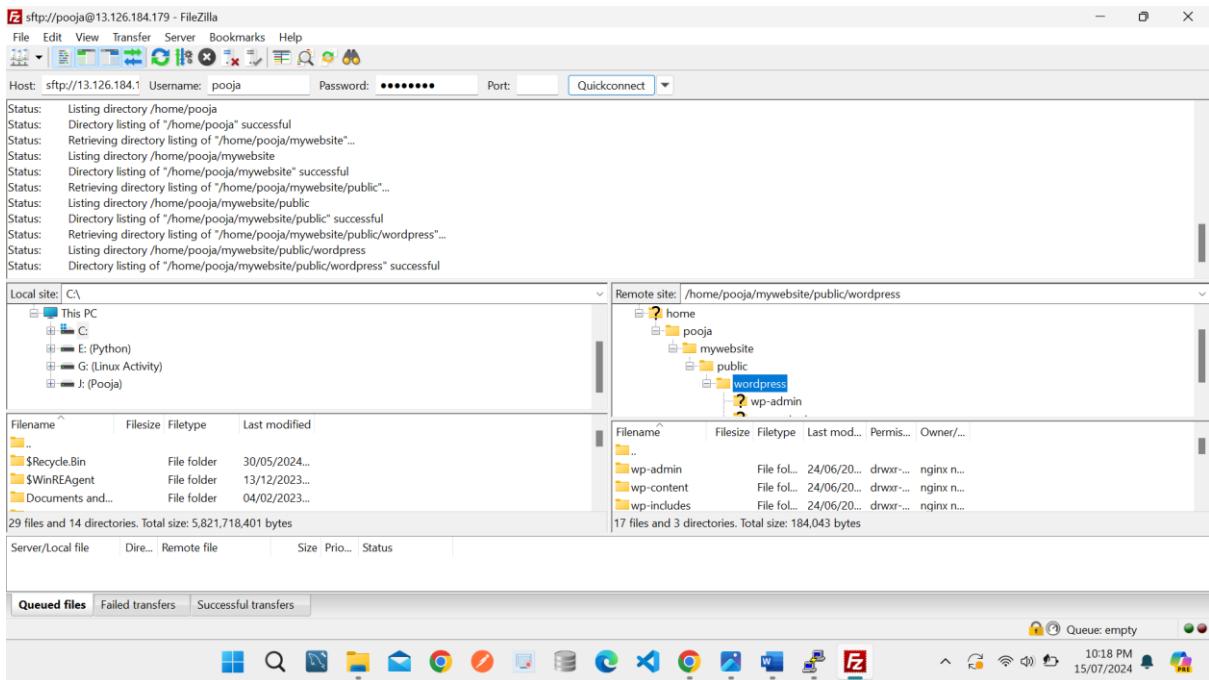
- Configuration details

```

[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]# cat /etc/ssh/sshd_config.d/sftp_pooja.conf
Match User pooja
ChrootDirectory /home/pooja
ForceCommand internal-sftp
AllowTcpForwarding no
PermitTunnel no
X11Forwarding no
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]# cd /etc/ssh/sshd_config.d/
[root@ip-172-31-7-11 sshd_config.d]# ls -lt
total 4
-rw-r--r-- 1 root root 127 Jul 14 20:49 sftp_pooja.conf
[root@ip-172-31-7-11 sshd_config.d]#
[root@ip-172-31-7-11 sshd_config.d]# 

```

- Open FileZilla and enter Host,username password and port 22 and Quickconnect we can check all the directories and files



Verify Via command (login to user using ssh and login sftp via password)

```
[root@ip-172-31-7-11 ~]# ssh pooja@13.233.150.53
The authenticity of host '13.233.150.53 (13.233.150.53)' can't be established.
ECDSA key fingerprint is SHA256:tuB3deNIULeH+0-ssy8Yn4Vv6J7yBGo2Sb3spKwFvD8.
ECDSA key fingerprint is MD5:df:39:f8:c1:53:04:b4:ee:f1:51:76:eb:d9:e59:1c.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '13.233.150.53' (ECDSA) to the list of known hosts.
pooja@13.233.150.53's password:
Permission denied, please try again.
pooja@13.233.150.53's password:
[Amazon Linux 2]
[Amazon Linux 2 End of Life is 2025-06-30.]
[A newer version of Amazon Linux is available!]
[Amazon Linux 2023, GA and supported until 2028-03-15.]
[https://aws.amazon.com/linux/amazon-linux-2023/]

No packages needed for security; 2 packages available
Run "sudo yum update" to apply all updates.
[pooja@ip-172-31-7-11 ~]$ [pooja@ip-172-31-7-11 ~]$ [pooja@ip-172-31-7-11 ~]$ [pooja@ip-172-31-7-11 ~]$ [pooja@ip-172-31-7-11 ~]$ [pooja@ip-172-31-7-11 ~]$ sftp -o StrictHostKeyChecking=no pooja@13.233.150.53
Warning: Permanently added '13.233.150.53' (ECDSA) to the list of known hosts.
pooja@13.233.150.53's password:
Connected to 13.233.150.53.
sftp> ls
mywebsite
sftp> cd mywebsite
sftp> ls
public
sftp> cd public
sftp> ls
wordpress
sftp> cd wordpress
sftp> ls
index.php      license.txt      readme.html      wp-activate.php      wp-admin      wp-blog-header.php
wp-comments-post.php  wp-config-sample.php  wp-config.php    wp-content      wp-cron.php      wp-includes
wp-load.php     wp-login.php       wp-mail.php      wp-settings.php   wp-signup.php   wp-trackback.php
wp-links-opml.php
wp-xmlrpc.php
```

4. Verify PhpMyAdmin

- phpMyAdmin login page can be view via <https://pooja.quest/phpmyadmin/index.php>
- Login can be done by username and password set for root
- Set Configuration details (# /etc/php-fpm.d/www.conf)
 - listen = /run/php-fpm/www.sock
 - php_value[session.save_path] = /var/lib/php/session

```
listen.owner = nginx
```

```
listen.group = nginx
```

```
listen.mode = 0660
```

A screenshot of a Windows desktop environment. In the center is a terminal window titled 'root@ip-172-31-1-251:~/ansible'. The window contains a block of PHP configuration code, specifically the 'php.ini' file. The code includes sections for FastCGI, session save paths, and session security settings. The terminal window has a dark background with white text. At the bottom of the screen is a taskbar with various icons for applications like File Explorer, Mail, and Google Chrome. The system tray shows the date (15/07/2024), time (10:08 PM), and battery level (5%).

```
; RPM: apache user chosen to provide access to the same directories as httpd
user = apache
; RPM: Keep a group allowed to write in log dir.
group = apache

; The address on which to accept FastCGI requests.
; Valid syntaxes are:
; 'ip.add.re.ss:port'      - to listen on a TCP socket to a specific IPv4 address on
;                           a specific port;
; '[ip:6:addr:ess]:port'   - to listen on a TCP socket to a specific IPv6 address on
;                           a specific port;
; 'port'                  - to listen on a TCP socket to all addresses
;                           (IPv6 and IPv4-mapped) on a specific port;
; '/path/to/unix/socket' - to listen on a unix socket.

; Note: This value is mandatory.
; Set listen(2) backlog.
; Default Value: 511
listen.backlog = 511

; Set permissions for unix socket, if one is used. In Linux, read/write
; permissions must be set in order to allow connections from a web server.
; Default Values: user and group are set as the running user
;                 mode is set to 0660

Listen = /run/php-fpm/www.sock
php_value[session.save_path] = /var/lib/php/session
listen.owner = nginx
listen.group = nginx
listen.mode = 0660

; When POSIX Access Control Lists are supported you can set them using
; these options, value is a comma separated list of user/group names.
; When set, listen.owner and listen.group are ignored
listen.acl_users = nginx
listen.acl_groups = nginx

; List of addresses (IPV4/IPv6) of FastCGI clients which are allowed to connect.
; Equivalent to the FCGI_WEB_SERVER_ADDRS environment variable in the original
; PHP FCGI (5.2.2+). Makes sense only with a tcp listening socket. Each address
; must be separated by a comma. If this value is left blank, connections will be
; accepted from any ip address.
; Default Value: any
listen.allowed_clients = 127.0.0.1
-- INSERT --
```

```
#vi /etc/phi.ini
```

```
Session.save_path="/var/lib/php/session"
```

A screenshot of a Windows desktop environment. In the center is a terminal window titled 'root@ip-172-31-1-251:~/ansible'. The window contains a block of PHP configuration code, specifically the 'php.ini' file. The code includes sections for session storage and cookie handling. The terminal window has a dark background with white text. At the bottom of the screen is a taskbar with various icons for applications like File Explorer, Mail, and Google Chrome. The system tray shows the date (15/07/2024), time (6:44 PM), and battery level (75%).

```
; where data files are stored. Note Windows users have to change this
; variable in order to use PHP's session functions.
;

; The path can be defined as:
;
; session.save_path = "/var/lib/php/session"

; where N is an integer. Instead of storing all the session files in
; /path, what this will do is use subdirectories N-levels deep, and
; store the session data in those directories. This is useful if
; your OS has problems with many files in one directory, and is
; a more efficient layout for servers that handle many sessions.
;

; NOTE 1: PHP will not create this directory structure automatically.
;         You can use the script in the ext/session dir for that purpose.
; NOTE 2: See the section on garbage collection below if you choose to
;         use subdirectories for session storage
;

; The file storage module creates files using mode 600 by default.
; You can change that by using
;
session.save_path = "/var/lib/php/session"
;

; where MODE is the octal representation of the mode. Note that this
; does not overwrite the process's umask.
; http://php.net/session.save-path

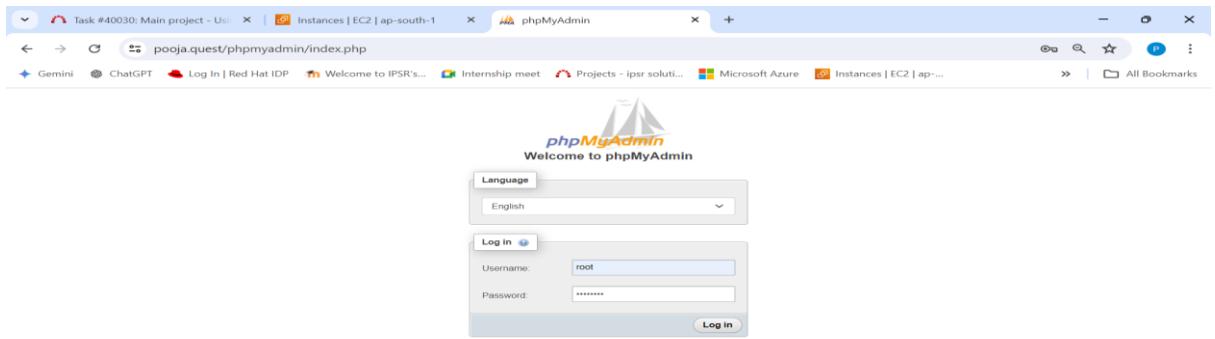
; RPM note : session directory must be owned by process owner
; for mod_php, see /etc/httpd/conf.d/php.conf
; for php-fpm, see /etc/php-fpm.d/*conf
;session.save_path = "/tmp"
;

; Whether to use strict session mode.
; Strict session mode does not accept uninitialized session ID and regenerate
; session ID if browser sends uninitialized session ID. Strict mode protects
; applications from session fixation via session adoption vulnerability. It is
; disabled by default for maximum compatibility, but enabling it is encouraged.
; https://wiki.php.net/rfc/strict_sessions
session.use_strict_mode = 0

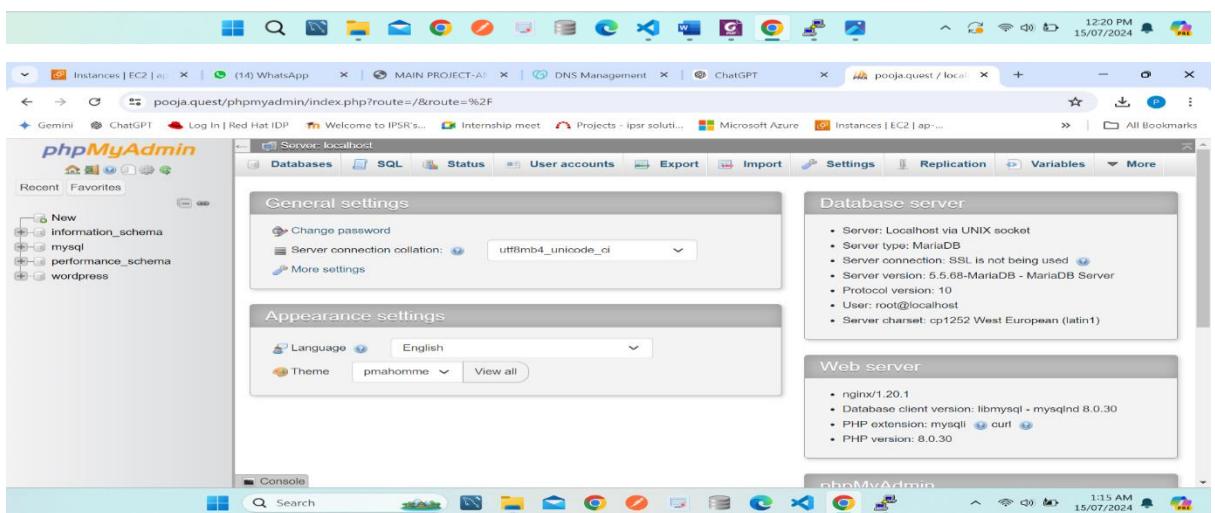
; whether to use cookies.
; http://php.net/session.use-cookies
session.use_cookies = 1
-- INSERT --
```

Output

phpMyAdmin -Login Page

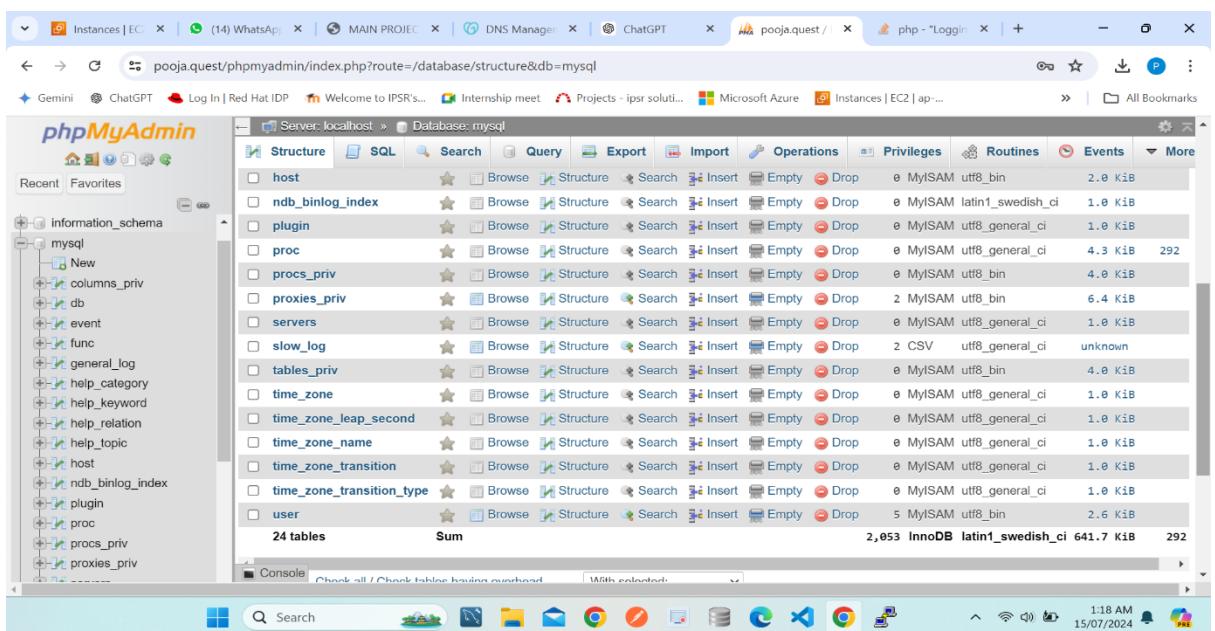


The screenshot shows the phpMyAdmin login interface. At the top, there's a header bar with various links like Gemini, ChatGPT, Log In | Red Hat IDP, Welcome to IPSR's..., Internship meet, Projects - ipsr solut..., Microsoft Azure, Instances | EC2 | ap..., and All Bookmarks. Below the header is the phpMyAdmin logo and the text "Welcome to phpMyAdmin". A language dropdown is set to English. The main area contains a "Log in" form with "Username" set to "root" and "Password" masked. A "Log in" button is at the bottom right of the form.



The screenshot shows the phpMyAdmin dashboard for "Server: localhost". On the left, there's a sidebar with "Recent" and "Favorites" sections, and a tree view showing databases: New, information_schema, mysql, performance_schema, and wordpress. The main content area has several tabs: Databases, SQL, Status, User accounts, Export, Import, Settings, Replication, Variables, and More. Under "General settings", it shows "Change password", "Server connection collation" set to utf8mb4_unicode_ci, and "More settings". Under "Appearance settings", it shows "Language" set to English, "Theme" set to pmahomme, and a "View all" link. Under "Database server", it lists server details: Localhost via UNIX socket, MariaDB, SSL not being used, version 5.5.68-MariaDB, protocol 10, user root@localhost, and character set cp1252 West European (latin1). Under "Web server", it lists nginx/1.20.1, libmysqld 8.0.30, mysqli, curl, and PHP version 8.0.30.

Mysql database



The screenshot shows the phpMyAdmin interface for the "mysql" database on "Server: localhost". The left sidebar shows the database tree with "mysql" selected. The main content area displays the structure of the "mysql" database, specifically the "host" table. The table has columns: host, user, db, and plugin. There are 24 tables in total, with a sum of 2,053 rows and 641.7 KiB of data. The table structure includes "Browse", "Structure", "Search", "Insert", "Empty", "Drop", and "Operations" buttons for each row. The "host" table has 2 rows, with a sum of 2 rows and 2.0 KiB of data. Other tables listed include ndb_binlog_index, plugin, proc, procs_priv, proxies_priv, servers, slow_log, tables_priv, time_zone, time_zone_leap_second, time_zone_name, time_zone_transition, time_zone_transition_type, and user.

Permissions

```
[root@ip-172-31-7-11 ~]# ls -ld /home/pooja
drwx---x 4 pooja pooja 112 Jul 15 08:01 /home/pooja
[root@ip-172-31-7-11 ~]# ls -ld /home/pooja/mywebsite
drwxr-xr-x 3 pooja pooja 20 Jul 14 20:48 /home/pooja/mywebsite
[root@ip-172-31-7-11 ~]# ls -ld /home/pooja/mywebsite/public
drwxr-xr-x 3 pooja pooja 23 Jul 14 20:52 /home/pooja/mywebsite/public
[root@ip-172-31-7-11 ~]# ls -ld /home/pooja/mywebsite/public/wordpress
drwxr-xr-x 5 nginx nginx 4096 Jul 14 22:35 /home/pooja/mywebsite/public/wordpress
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]#
```

- Open a web browser and complete WordPress installation through the web interface:

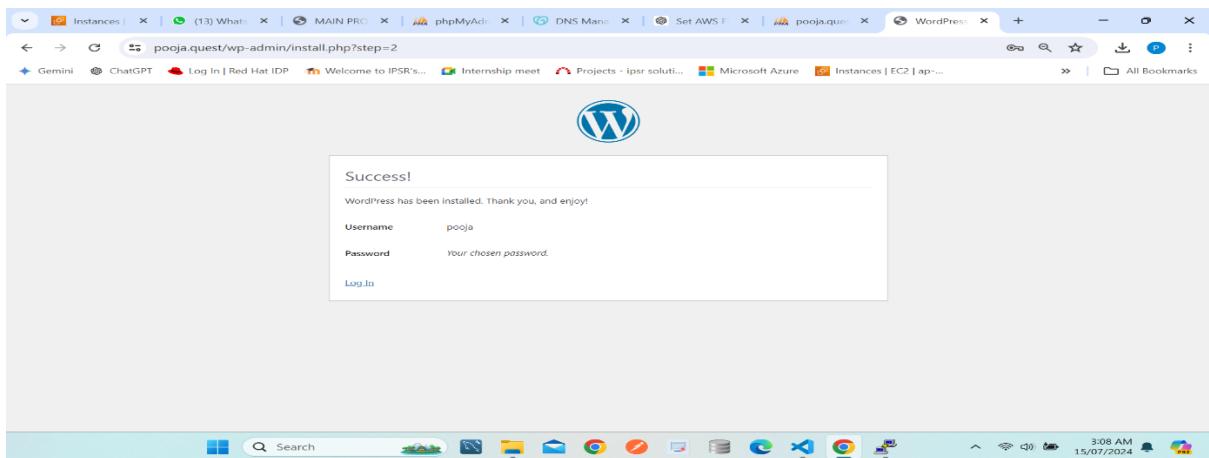
<http://pooja.quest>

You should see the WordPress setup page where you can enter your site title, username, password, and email. And verify the Installation

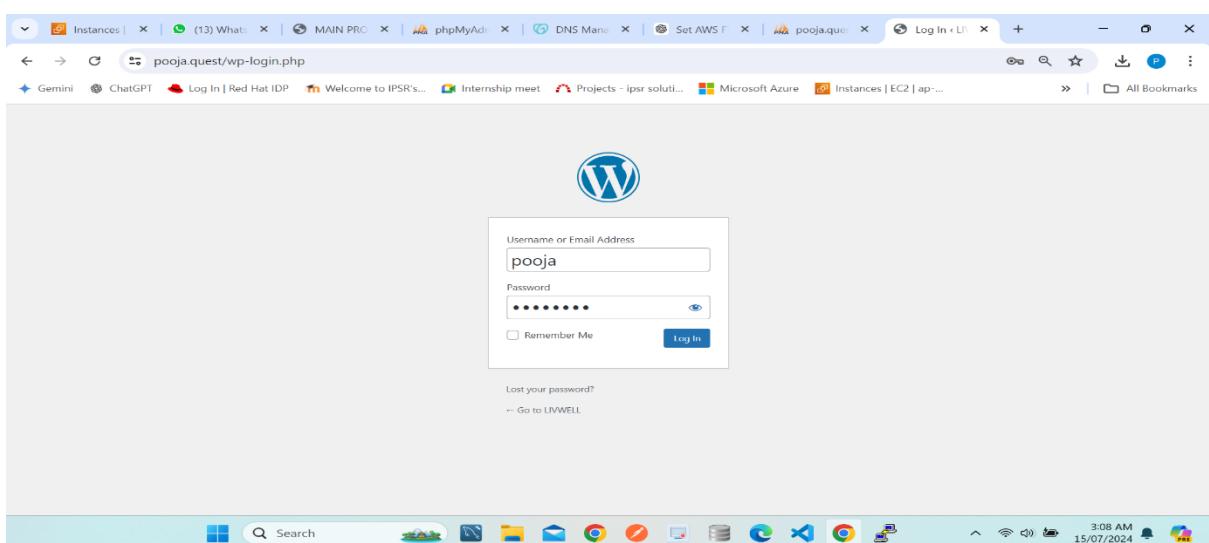
Once you complete the setup, you should be able to log in to the WordPress admin dashboard via

<http://pooja.quest/wp-admin>

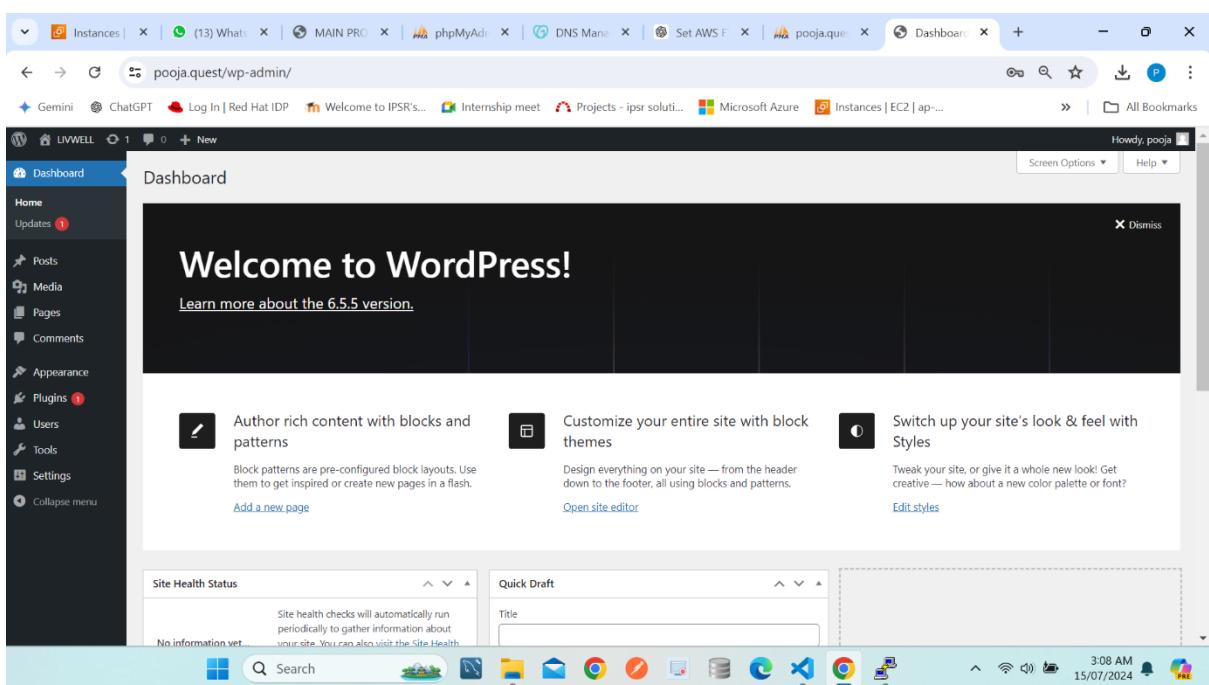




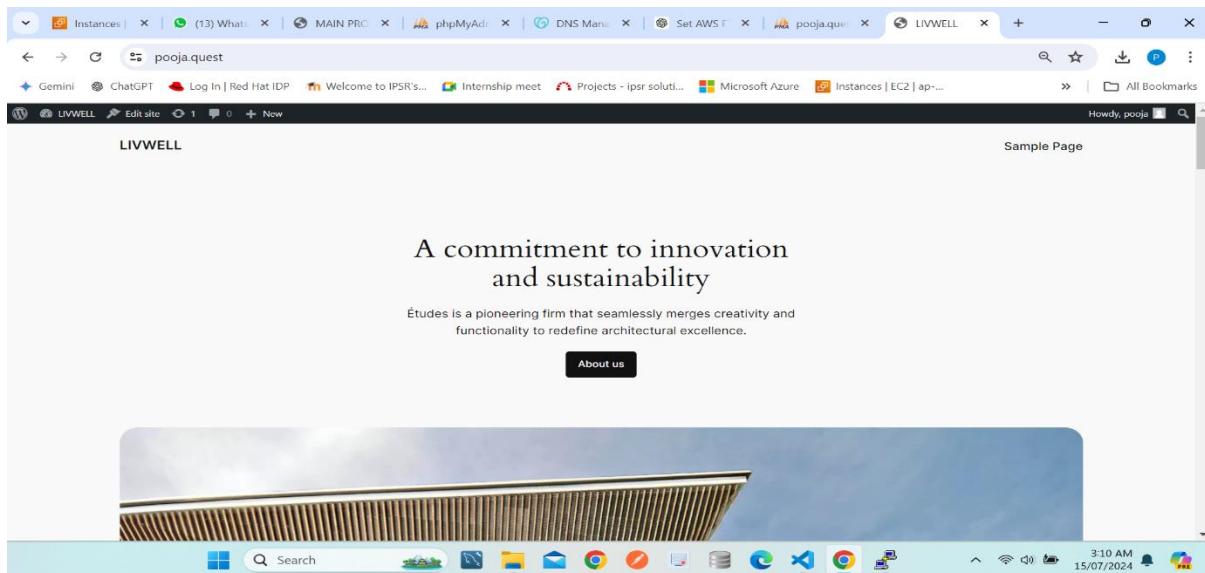
Login Page



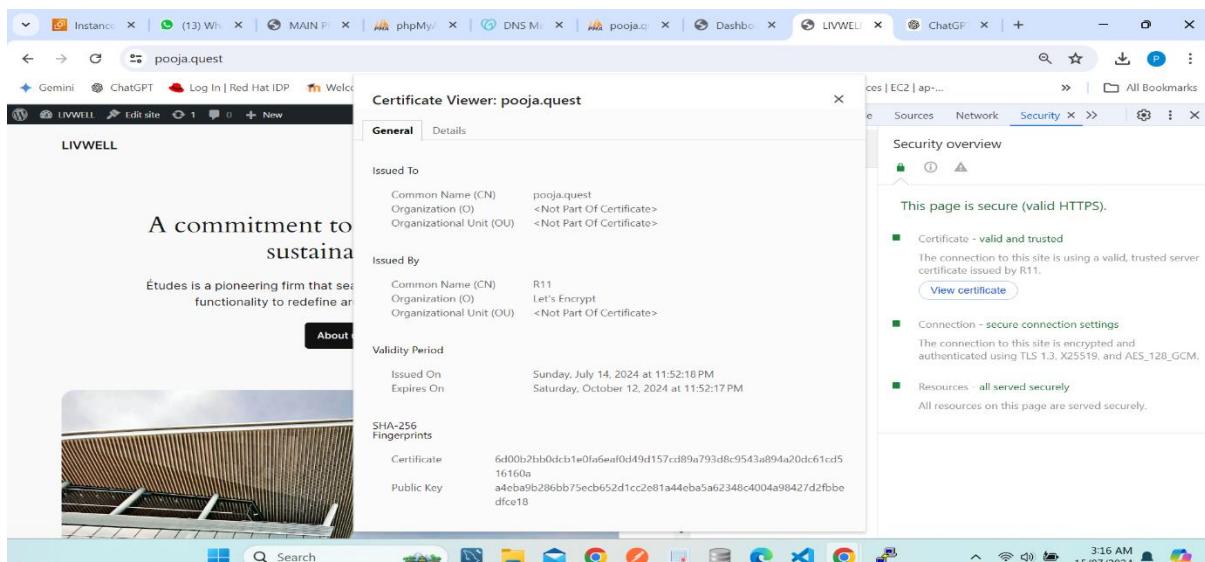
Admin dashboard



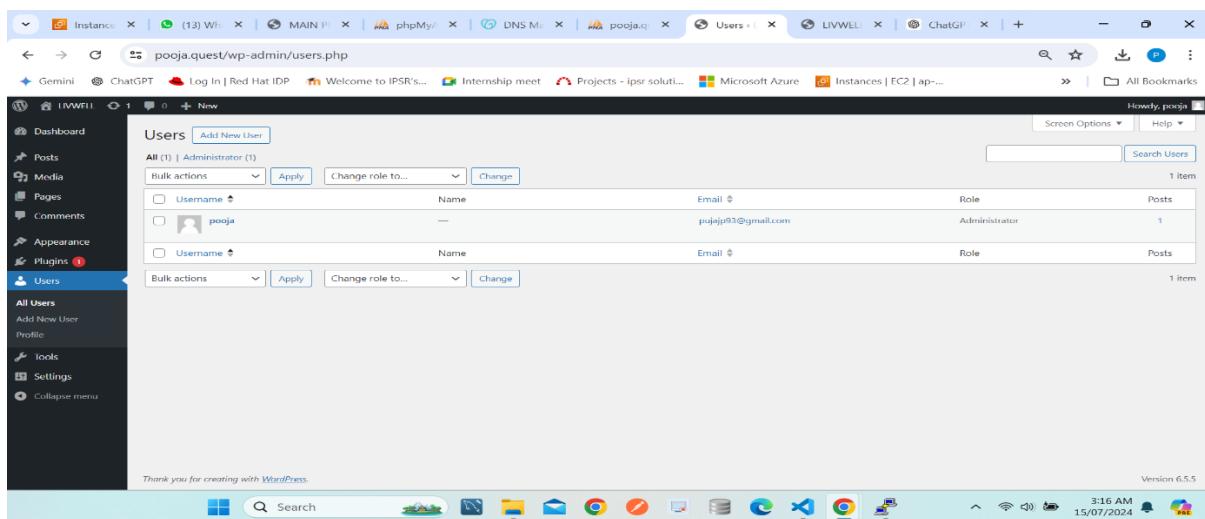
Website



Website ssl certification



User Created



Wordpress Database

Recent | Favorites

Filters

Containing the word:

Table	Action	Rows	Type	Collation	Size	Overhead
wp_commentsmeta	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48.0 KB	-
wp_comments	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_unicode_ci	96.0 KB	-
wp_links	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
wp_options	Browse Structure Search Insert Empty Drop	144	InnoDB	utf8mb4_unicode_ci	144.0 KB	-
wp_postmeta	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_unicode_ci	48.0 KB	-
wp_posts	Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_unicode_ci	80.0 KB	-
wp_termmeta	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48.0 KB	-
wp_terms	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_unicode_ci	48.0 KB	-
wp_term_relationships	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
wp_term_taxonomy	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_unicode_ci	48.0 KB	-
wp_usermeta	Browse Structure Search Insert Empty Drop	18	InnoDB	utf8mb4_unicode_ci	48.0 KB	-
wp_users	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_unicode_ci	64.0 KB	-
12 tables	Sum	174	InnoDB	latin1_swedish_ci	736.0 KB	0 B

[Up](#) [Check all](#) [With selected...](#)

Data dictionary
Console

Showing rows 0 - 4 (5 total). Query took 0.00002 seconds.

Host	User	Password	Select_priv	Insert_priv	Update_priv	Delete_priv	Create_priv	Drop_priv	Reload_priv
ip-172-31-7-11.ap	root	*4A6C1EF614FDDFD6ED47028CB0A0E1A0D6283F30	Y	Y	Y	Y	Y	Y	Y
1.internal.com	root	*4A6C1LH184B1DH-D6E47028CB0A0E1A0B6283F30	Y	Y	Y	Y	Y	Y	Y
127.0.0.1	root	*4A0C1EF614FBDFD6ED47028CB0A0E1A0B6283F30	Y	Y	Y	Y	Y	Y	Y
localhost	root	*4A6C1LH184B1DH-D6E47028CB0A0E1A0D6283F30	N	N	N	N	N	N	N
wordpress	root	*4A6C1EF614FBDFFD6ED47028CB0A0E1A0B6283F30	N	N	N	N	N	N	N

Query results operations:

Console Copy to clipboard Export Display chart Create view

SSL Certification Check Via Terminal

```

root@ip-172-31-7-11:~
lrwxrwxrwx 1 root root 49 Jul 9 23:43 ca.bundle.crt -> /etc/pki/ca_trust/extracted/csm/tls_ca_bundle.pem
lrwxrwxrwx 1 root root 55 Jul 9 23:43 ca-bundle.trust.crt -> /etc/pki/ca-trust/extracted/openssl/ca-bundle.trust.crt
-rw-r--r-- 1 root root 610 Feb 15 22:26 make_dummy_cert
-rw-r--r-- 1 root root 2516 Feb 15 22:26 Makefile
-rw-r--r-x 1 root root 829 Feb 15 22:26 renew-dummy-cert
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]# sudo find / -name "*.crt"
/etc/pki/ca-trust/source/ca-bundle.legacy.crt
/etc/pki/tls/certs/ca-bundle.crt
/etc/pki/tls/certs/ca-bundle.trust.crt
/usr/lib/python2.7/site-packages/ndg/httpsclient/test/pki/localhost.crt
/usr/share/pki/ca-trust-legacy/ca-bundle.legacy.default.crt
/usr/share/pki/ca-trust-legacy/ca-bundle.legacy.disabled.crt
/home/pooja/mywebsite/public/wordpress/wp-includes/certificates/ca-bundle.crt
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]# grep -r "ssl_certificate" /etc/nginx/
/etc/nginx/conf.d/pooja.quest.conf:    ssl_certificate /etc/letsencrypt/live/pooja.quest/fullchain.pem; # managed by Certbot
/etc/nginx/conf.d/pooja.quest.conf:    ssl_certificate_key /etc/letsencrypt/live/pooja.quest/privkey.pem; # managed by Certbot
/etc/nginx/nginx.com.conf.default:    #    ssl_certificate      cert.pem;
/etc/nginx/nginx.com.conf.default:    #    ssl_certificate_key      cert.key;
/etc/nginx/nginx.conf:    ssl_certificate      "/etc/pki/nginx/server.crt";
/etc/nginx/nginx.conf:    ssl_certificate_key      "/etc/pki/nginx/private/server.key";
[root@ip-172-31-7-11 ~]# grep -r "SSLCertificateFile" /etc/apache2/
grep: /etc/apache2/: No such file or directory
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]# openssl x509 -in /etc/ssl/certs/pooja.quest.crt -text -noout
Error opening Certificate /etc/ssl/certs/pooja.quest.crt
14080588604632:error:02001002:system library:fopen('etc/ssl/certs/pooja.quest.crt','r')
14080588604632:error:20074002:BIO routines:FILE_CTRL:system lib:bss_file.c:404:
unable to load certificate
[root@ip-172-31-7-11 ~]#
[root@ip-172-31-7-11 ~]#

```

This is the Blog Created check via url <https://pooja.quest/2024/07/14/livwell/>

LIVWELL

Livwell

Jul 14, 2024 — by pooja in Uncategorized

Made with simple, whole food ingredients. Nutritionist crafted, our sauces are nutrient dense, veggie packed, and keep your family nourished and full.

CRAFTED WITH ONLY THE FINEST, FRESHEST INGREDIENTS

Plant-based nutrient-dense superfood sauces to create foods only limited by your imagination.

Sample Page

Login Credentials

1. SFTP

Host: 13.126.184.179 (Public IP)

Username: pooja

Password: pooja\$#1

2. PhpMyAdmin

URL: <https://pooja.quest/phpmyadmin/index.php>

Username: root

Password: "pooja\$#!

3. WordPress admin dashboard

URL: <https://pooja.quest/wp-login.php>

Username: pooja

Password: "pooja\$#!

URL for sample site: <https://pooja.quest/>

URL for Blog: <https://pooja.quest/2024/07/14/livwell/>