Bytexl Guided project

Prerequisites

1. Ubuntu Server (18.04 or later).
2. SSH access to your server (or terminal access if it's local).
3. Domain name (optional, but recommended).
4. Basic knowledge of Linux commands.

Step 1: Update and Upgrade the System

First, update your system packages to the latest versions:

```
sudo apt update
sudo apt upgrade -y
```

Step 2: Install Apache Web Server

WordPress needs a web server to serve the site. We'll use Apache:

```
sudo apt install apache2 -y
```

Once installed, enable and start Apache:

```
sudo systemctl enable apache2
sudo systemctl start apache2
```

To verify if Apache is working, open your browser and enter your server's IP address. You should see the Apache default page.

Step 3: Install MySQL Database

WordPress needs a database to store its data. Install MySQL:

sudo apt install mysql-server -y

Secure the MySQL installation:

sudo mysql_secure_installation

This script will ask you to configure several security settings:

- Set the MySQL root password.
- Remove anonymous users.
- Disallow root login remotely.
- Remove test databases.
- Reload privilege tables.

Once finished, log into the MySQL shell:

sudo mysql -u root -p

Now, create a database and a user for WordPress:

```
CREATE DATABASE wordpress_db;
CREATE USER 'wp_user'@'localhost' IDENTIFIED BY 'your_strong_password';
GRANT ALL PRIVILEGES ON wordpress_db.* TO 'wp_user'@'localhost';
FLUSH PRIVILEGES;
EXIT;
```

Step 4: Install PHP

WordPress is written in PHP, so you need to install PHP and its extensions:

sudo apt install php libapache2-mod-php php-mysql php-xml php-gd php-mbstring php-curl php-zip php-imagick -y

Step 5: Configure Apache for WordPress

Now, create a virtual host file for your WordPress site. This is especially useful if you're running multiple sites on your server.

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

In this file, add the following configuration:

```
apache
<VirtualHost *:80>
    ServerAdmin admin@example.com
    DocumentRoot /var/www/wordpress
    ServerName example.com
    ServerAlias www.example.com

    <Directory /var/www/wordpress/>
        Options FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

- Replace `example.com` with your domain name (or leave it as the server's IP address for now).

Save and exit the file.

Enable the site and rewrite module for permalinks:

```
sudo a2ensite wordpress.conf
sudo a2enmod rewrite
sudo systemctl restart apache2
```

Step 6: Download and Install WordPress

Go to the `/tmp` directory and download the latest WordPress:

```
cd /tmp
wget https://wordpress.org/latest.tar.gz
```

Extract the WordPress archive:

```
tar -xvzf latest.tar.gz
```

Move the WordPress files to the web root directory:

```
sudo mv wordpress /var/www/wordpress
```

Set proper permissions on the WordPress directory:

```
sudo chown -R www-data:www-data /var/www/wordpress
sudo chmod -R 755 /var/www/wordpress
```

Step 7: Configure WordPress

Now, you need to configure WordPress to use the MySQL database. First, go to the WordPress directory:

```
cd /var/www/wordpress
```

Copy the sample configuration file to `wp-config.php`:

```
cp wp-config-sample.php wp-config.php
```

Edit the file to add your database credentials:

sudo nano wp-config.php

Look for these lines and replace them with your database information:

php
define('DB_NAME', 'wordpress_db');
define('DB_USER', 'wp_user');
define('DB_PASSWORD', 'your_strong_password');
define('DB_HOST', 'localhost');

Save and exit the file.

Step 8: Complete WordPress Installation via Web Browser

Open your browser and go to `http://your_server_ip_or_domain`. You should be redirected to the WordPress installation page.

1. Choose Language.
2. fill out the website details(site name, admin username, password, email, etc.).
3. Click Install WordPress.

Once done, you can log in to the WordPress admin dashboard at `http://your_server_ip_or_domain/wp-admin`.

Step 9: Configure Firewall (Optional)

If you're running a firewall (like `ufw`), allow HTTP and HTTPS traffic:

sudo ufw allow 'Apache Full'

step 10: Install SSL (Optional but Recommended)

Then, obtain and install the SSL certificate:

sudo certbot --apache

Follow the prompts, and Certbot will automatically configure SSL for your site. You can test your SSL by visiting `https://your_domain`.