

WordPress Business Website Server with SSL and Backup Tools

Customer Instructions by Code Creator

Ubuntu 24.04 LTS | Docker Compose | WordPress | MariaDB | Redis | Nginx | Optional SSL

This product launches a ready to use WordPress website server with a public IP first setup, MariaDB database, Redis cache support, Nginx reverse proxy, backup helper, optional Let's Encrypt SSL helper, and first boot guidance.

1. Quick Start

Connect to your instance using your Amazon private key and the **ubuntu** user.

After you connect by SSH, read the first login notes:

```
cat /home/ubuntu/FIRST_LOGIN.txt
```

Show the server URLs at any time:

```
codecreator-wordpress-url
```

Check the current application status:

```
codecreator-wordpress-status
```

2. Required AWS Security Group Ports

| Port | Protocol | Purpose |
|------|----------|--|
| 22 | TCP | SSH access using your Amazon private key and the ubuntu user |
| 80 | TCP | WordPress setup, HTTP access, and the Code Creator help page |
| 443 | TCP | HTTPS access after optional domain SSL setup |

3. First WordPress Setup

1. Open the WordPress setup URL shown in `/home/ubuntu/FIRST_LOGIN.txt` or by the `codecreator-wordpress-url` command.
2. Choose the site language.
3. Create your WordPress administrator username and password.
4. Save the username and password you create. Code Creator does not store or know your WordPress admin password.
5. Log in to the WordPress administrator dashboard and begin configuring your site.

Your public IP URL will look like this, but with your own instance IP address:

`http://YOUR_PUBLIC_IP/`

4. Code Creator Help Page

The product includes a browser based help page with your current public IP, useful commands, setup notes, backup guidance, and optional SSL instructions.

`http://YOUR_PUBLIC_IP/codecreator.html`

5. Recommended After WordPress Setup: Enable Redis Object Cache

After you complete the WordPress web setup and create your WordPress administrator account, enable Redis object caching for better WordPress performance:

`sudo codecreator-wordpress-enable-redis`

The helper checks that WordPress is installed, installs and activates the Redis Object Cache plugin, writes the correct Docker Redis host settings, and enables the Redis object cache drop in.

6. Create a Backup

Create a backup of the WordPress database and WordPress files with this command:

`sudo codecreator-wordpress-backup`

Backups are saved in:

`/opt/codecreator-wordpress/backups`

Each backup archive includes:

- `wordpress-database.sql`
- `wordpress-files.tar.gz`
- `README.txt`

7. Optional Domain Name and SSL Setup

The server works first by public IP. A domain name is optional. To enable HTTPS with Let's Encrypt, first point your domain DNS A record to this instance public IP.

After DNS is correct, run:

`sudo codecreator-wordpress-ssl yourdomain.com admin@yourdomain.com`

Replace `yourdomain.com` with your domain and `admin@yourdomain.com` with your email address. The helper updates Nginx, requests a Let's Encrypt certificate, redirects HTTP to HTTPS, and attempts to update WordPress home and siteurl if WordPress is already installed.

8. Helpful Commands

| Command | Purpose |
|---|---|
| <code>cat /home/ubuntu/FIRST_LOGIN.txt</code> | Read first login notes and URLs |
| <code>codecreator-wordpress-url</code> | Show public WordPress and help page URLs |
| <code>codecreator-wordpress-status</code> | Check Docker, Nginx, WordPress, MariaDB, Redis, and HTTP status |
| <code>sudo codecreator-wordpress-enable-redis</code> | Enable Redis object cache after WordPress setup |
| <code>sudo codecreator-wordpress-backup</code> | Create a backup of the WordPress database and files |
| <code>sudo codecreator-wordpress-ssl yourdomain.com admin@yourdomain.com</code> | Enable HTTPS after your domain DNS points to the instance |

9. Common File Locations

| Path | Description |
|--|---|
| <code>/opt/codecreator-wordpress</code> | Main application directory |
| <code>/opt/codecreator-wordpress/docker-compose.yml</code> | Docker Compose stack definition |
| <code>/opt/codecreator-wordpress/backups</code> | Backup archive location |
| <code>/home/ubuntu/FIRST_LOGIN.txt</code> | First login notes generated on first boot |

10. Recommended Instance Sizes and Storage

| Use Case | Recommended Instance | Root Storage |
|-----------------------|--------------------------------|---------------------|
| Small test site | t3.small or t3.medium | 30 GB gp3 minimum |
| Recommended default | t3.medium or t3.large | 30 to 50 GB gp3 |
| Heavier business site | t3.large, m6i.large, or larger | 50 GB gp3 or larger |

11. Troubleshooting

Check status first:

```
codecreator-wordpress-status
```

Restart the stack if needed:

```
cd /opt/codecreator-wordpress
```

```
sudo docker compose up -d
```

View container status directly:

```
cd /opt/codecreator-wordpress
```

```
sudo docker compose ps
```

View recent WordPress container logs:

```
cd /opt/codecreator-wordpress
```

```
sudo docker compose logs --tail=100 wordpress
```

12. Important Notes

- The server is designed to work by public IP first. A domain is optional.
- Do not lose the WordPress admin username and password you create during setup.
- For production use, schedule regular backups and consider copying backup archives to external storage such as Amazon S3.
- Keep WordPress themes and plugins updated through the WordPress administrator dashboard.
- Open only the required AWS security group ports for your use case.