

# Complete Guide: Self-Hosted WordPress with Docker + Cloudflare Tunnel

## 1. System Overview

This setup creates a fully self-hosted WordPress instance using Docker, with public access through Cloudflare Tunnel (no port forwarding).

Architecture: Internet → Cloudflare → Tunnel → localhost → Docker → MySQL → Disk

## 2. Requirements

- Debian 13 (trixie)
- Domain (bmiller8008.com)
- Storage mounted at /mnt/MediaDrive
- Cloudflare account

## 3. Install Docker

```
sudo apt update
sudo apt install docker.io docker-compose-plugin -y
sudo systemctl enable docker
sudo systemctl start docker
sudo usermod -aG docker $USER
```

## 4. Create Project Directory

```
mkdir -p /mnt/MediaDrive/docker/wordpress
cd /mnt/MediaDrive/docker/wordpress
```

## 5. Docker Compose File

```
services:
  db:
    image: mysql:8.4
    container_name: wordpress-db
    restart: unless-stopped
    environment:
      MYSQL_DATABASE: wordpress
      MYSQL_USER: wordpress
      MYSQL_PASSWORD: change_this_wp_password
      MYSQL_ROOT_PASSWORD: change_this_root_password
    volumes:
      - /mnt/MediaDrive/docker/wordpress/mysql:/var/lib/mysql
    healthcheck:
      test: ["CMD-SHELL", "mysqladmin ping -h 127.0.0.1 -uroot -p$$MYSQL_ROOT_PASSWORD || exit 1"]
      interval: 10s
      timeout: 5s
```

```
    retries: 12
    start_period: 40s

wordpress:
  image: wordpress:latest
  container_name: wordpress-app
  restart: unless-stopped
  depends_on:
    db:
      condition: service_healthy
  environment:
    WORDPRESS_DB_HOST: db:3306
    WORDPRESS_DB_NAME: wordpress
    WORDPRESS_DB_USER: wordpress
    WORDPRESS_DB_PASSWORD: change_this_wp_password
  ports:
    - "8081:80"
  volumes:
    - /mnt/MediaDrive/docker/wordpress/html:/var/www/html
```

## 6. Critical Gotcha (MySQL 8.4)

Do NOT use:

```
command: --default-authentication-plugin=mysql_native_password
```

This breaks MySQL 8.4 and causes restart loops and missing system tables.

## 7. Clean Initialization

```
docker compose down
sudo rm -rf /mnt/MediaDrive/docker/wordpress/mysql
sudo rm -rf /mnt/MediaDrive/docker/wordpress/html
mkdir -p /mnt/MediaDrive/docker/wordpress/mysql
mkdir -p /mnt/MediaDrive/docker/wordpress/html
sudo chown -R 999:999 /mnt/MediaDrive/docker/wordpress/mysql
```

## 8. Start Services

```
docker compose up -d
```

## 9. Verify Locally

```
curl -I http://localhost:8081
```

## 10. Install Cloudflare Tunnel

```
sudo apt install cloudflared
cloudflared tunnel login
cloudflared tunnel create wordpress
```

## 11. Configure Tunnel

```
sudo nano /etc/cloudflared/config.yml

tunnel: YOUR_TUNNEL_ID
credentials-file: /etc/cloudflared/YOUR_TUNNEL_ID.json

ingress:
  - hostname: bmiller8008.com
    service: http://localhost:8081
  - service: http_status:404
```

## 12. Start Tunnel

```
sudo systemctl enable cloudflared
sudo systemctl restart cloudflared
```

## 13. Access Site

Visit: <https://bmiller8008.com>

## 14. WordPress Setup

Complete installer and login at /wp-admin

## 15. Fix HTTPS Reverse Proxy

```
if (isset($_SERVER['HTTP_X_FORWARDED_PROTO']) && $_SERVER['HTTP_X_FORWARDED_PROTO'] === 'https') {
    $_SERVER['HTTPS'] = 'on';
}
```

## 16. Backup Strategy

Backup these directories regularly:

```
/mnt/MediaDrive/docker/wordpress/mysql
/mnt/MediaDrive/docker/wordpress/html
```

## 17. Debugging Checklist

```
docker ps
docker logs wordpress-db
docker logs wordpress-app
ss -lntp
journalctl -u cloudflared -n 50
```

## 18. Common Issues

- MySQL restart loop → remove bad auth plugin + wipe DB
- Cloudflare 502 → wrong port (must match Docker host port)

- WordPress DB error → wrong DB host (must be db:3306)