

12: WordPress - Production-Ready Setup

WordPress is a widely used CMS for websites and blogs. This guide covers installation and runtime specifics when deploying via Infinity Tools. For platform usage, administration, and theme/plugin development, refer to the [official WordPress documentation](#).

Architecture Overview

- **Services:** WordPress (PHP/Apache), MariaDB 10.11, optional Redis cache
- **Reverse Proxy:** Traefik with Let's Encrypt (recommended) or standalone
- **Networks:** Traefik proxy network + `borgmatic-db` for DB backups
- **Multi-instance:** Supported via `--instance=<name>`

Prerequisites

- **Traefik installed** (Chapter 4)
- **Docker installed** (Chapter 3)
- **Apprise installed** (Chapter 5) for notifications
- **Borgmatic installed** (Chapter 6) for automated backups
- **Domain configured** (Chapter 4.5) for HTTPS

Interdependencies: MariaDB is joined to `borgmatic-db` for backup discovery. Borgmatic depends on Apprise for notifications.

Installation Methods

Via Infinity Tools Menu

```
☐☐APPLICATIONS → WordPress → Install
```

Command Line

```
# Status (no changes)
sudo bash /opt/InfinityTools/Solutions/setup-wordpress.sh --status

# Default instance (interactive)
sudo bash /opt/InfinityTools/Solutions/setup-wordpress.sh --install

# Named instance
sudo bash /opt/InfinityTools/Solutions/setup-wordpress.sh --install --instance=blog2
```

Key Configuration

- **SSL Mode:** Traefik (HTTPS) or standalone (HTTP or self-signed HTTPS)
- **Domain:** Required for Traefik (e.g., `myblog.com`)
- **Standalone Port:** Required if not using Traefik
- **Redis Cache:** Optional. Can be enabled during install
- **Multi-Instance Paths:**
 - Default: `/opt/speedbits/wordpress`
 - Instance `blog2`: `/opt/speedbits/wordpress-blog2`

Generated Files & Directories

- `$WP_DIR/wp_data/` — WordPress files
- `$WP_DIR/db_data/` — MariaDB data
- `$WP_DIR/redis_data/` — Redis persistence (if enabled)
- `$WP_DIR/docker-compose.yml` — Service definition
- `$WP_DIR/php/uploads.ini` — PHP upload limits (50MB)
- `$WP_DIR/db_password.txt` — Generated DB password

Traefik Mode (Highlights)

```
services:
  db:
    image: mariadb:10.11
    networks: [ ${NETWORK}, borgmatic-db ]

  redis:          # if enabled
    image: redis:7-alpine
```

```
command: redis-server --maxmemory 64mb --maxmemory-policy allkeys-lru
```

```
wordpress:
```

```
image: wordpress:latest
```

```
environment:
```

```
WORDPRESS_DB_HOST: wp-db:3306
```

```
WORDPRESS_DB_USER: wpuser
```

```
WORDPRESS_DB_PASSWORD: ${FROM_FILE}
```

```
WORDPRESS_DB_NAME: wordpress
```

```
WORDPRESS_TABLE_PREFIX: wp_
```

```
WORDPRESS_CONFIG_EXTRA: |
```

```
define('DISALLOW_FILE_EDIT', true);
```

```
define('FORCE_SSL_ADMIN', true);
```

```
define('WP_MEMORY_LIMIT', '512M');
```

```
define('WP_MAX_MEMORY_LIMIT', '1024M');
```

```
define('WP_CACHE', true);
```

```
define('WP_POST_REVISIONS', 10);
```

```
define('AUTOSAVE_INTERVAL', 300);
```

```
define('WP_IMAGE_EDITORS', ['WP_Image_Editor_GD']);
```

```
labels:
```

```
- "traefik.enable=true"
```

```
- "traefik.http.routers.${ROUTER_NAME}.rule=Host(`${DOMAIN}`)"
```

```
- "traefik.http.routers.${ROUTER_NAME}.entrypoints=websecure"
```

```
- "traefik.http.routers.${ROUTER_NAME}.tls.certresolver=myresolver"
```

```
- "traefik.http.routers.${ROUTER_NAME}-www.rule=Host(`www.${DOMAIN}`)"
```

```
- "traefik.http.routers.${ROUTER_NAME}-www.middlewares=${ROUTER_NAME}-redirect"
```

```
- "traefik.http.middlewares.${ROUTER_NAME}-
```

```
redirect.redirectregex.regex=^https://www\.\.${DOMAIN}/(.*)"
```

```
- "traefik.http.middlewares.${ROUTER_NAME}-
```

```
redirect.redirectregex.replacement=https://${DOMAIN}/${1}"
```

```
- "traefik.http.middlewares.${ROUTER_NAME}-redirect.redirectregex.permanent=true"
```

Redis Object Cache (Recommended)

Enable Redis during installation (optional), then install the free **Redis Object Cache** plugin for significant performance gains.

- Dashboard → **Plugins** → **Add New**
- Search **Redis Object Cache** by Till Krüss

- Install → Activate → **Settings** → **Redis** → **Enable Object Cache**

[Redis Object Cache plugin \(wordpress.org\)](https://wordpress.org/plugins/redis-object-cache/)

Post-Install Hardening & Defaults

- File editing disabled; XML-RPC disabled
- OPcache enabled; memory limits tuned
- WWW → non-WWW redirect configured (Traefik)
- Let's Encrypt certificates via Traefik

Backup Integration (Borgmatic)

- MariaDB is auto-registered with Borgmatic (if available)
- Include in backups:
 - `$WP_DIR/wp_data`
 - `$WP_DIR/db_data`
 - `$WP_DIR/redis_data` (if enabled)
- Ensure Apprise notifications are configured

Operations

```
# Logs
docker logs wordpress
docker logs wp-db

# Restart
cd $WP_DIR && docker compose restart

# Update (safe)
cd $WP_DIR && docker compose pull && docker compose up -d

# Instance status
sudo bash /opt/InfinityTools/Solutions/setup-wordpress.sh --status

# Wipe and reinstall (destructive)
sudo bash /opt/InfinityTools/Solutions/setup-wordpress.sh --install --deleteall
```

Troubleshooting

- **Database connection errors:** verify `db_password.txt`, check `wp-db` logs, ensure volumes mounted
- **SSL/ACME issues:** check Traefik logs, DNS A/AAAA, ports 80/443 open
- **Performance:** enable Redis Object Cache; review plugins/themes bloat
- **Uploads:** default max upload 50MB via `$WP_DIR/php/uploads.ini`

Security Best Practices

- Keep core, themes, and plugins updated
- Limit admin accounts; enforce strong passwords and 2FA
- Review plugin footprint; remove unused components
- Regularly test backups and restore procedures

Verification Checklist

- WordPress, MariaDB (and Redis if used) containers running
- Site reachable over HTTPS with valid cert (Traefik)
- Admin login working; permalinks saved
- Redis Object Cache enabled (if configured)
- Backups configured and successful

Revision #2

Created 4 November 2025 17:23:46 by bjoern

Updated 17 November 2025 16:37:07 by bjoern