

EasyEngine CLI Cheat Sheet



A quick-reference guide to the most common **site**, **WordPress**, **cron**, and **shell** workflows.

“ Conventions

1. Replace `example.com` with your domain (site name).
2. Square brackets `[...]` mean *optional*.
3. Angle brackets `<...>` mean *required*.
4. SSL values you'll see here: `le` (Let's Encrypt) and `self` (self-signed).

Site Management (All Site Types)



1) Create a site (HTML)

Command

```
ee site create --type=html <site-name> [--ssl=<value>] [--wildcard] [--skip-status-check]
```

Options

1. `<site-name>`: Domain / site name.
2. `--ssl=<value>`: Enable SSL.
 1. `le` = Let's Encrypt
 2. `self` = Self-signed
3. `--wildcard`: Request a wildcard certificate (typically with Let's Encrypt).
4. `--skip-status-check`: Skip checking site status before action.

Examples

1. Create an HTML site:

```
ee site create example.com --type=html
```

2. Create with Let's Encrypt SSL:

```
ee site create example.com --type=html --ssl=le
```

3. Create with Let's Encrypt wildcard SSL:

```
ee site create example.com --type=html --ssl=le --wildcard
```

4. Create with self-signed SSL:

```
ee site create example.com --type=html --ssl=self
```

2) Delete a site

Command

```
ee site delete <site-name> [--yes]
```

Options

1. `<site-name>`: Site to delete.
2. `--yes`: Skip confirmation prompt.

Example

```
ee site delete example.com
```

3) Update a site (commonly used for SSL)



Command

```
ee site update [<site-name>] [--ssl=<ssl>] [--wildcard]
```

Options

1. `[<site-name>]`: Site to update (if omitted, behavior depends on EE context/version).
2. `--ssl=<ssl>`: Enable SSL on an existing site.
3. `--wildcard`: Enable wildcard SSL.

Examples

1. Add Let's Encrypt SSL:

```
ee site update example.com --ssl=le
```

2. Add wildcard Let's Encrypt SSL:

```
ee site update example.com --ssl=le --wildcard
```

3. Add self-signed SSL:

```
ee site update example.com --ssl=self
```

4) Enable / Disable a site

Enable (starts containers if stopped)

```
ee site enable [<site-name>] [--force] [--verify]
```

Options

1. `--force`: Force execution.
2. `--verify`: Verify dependent global services (slower, but safer).

Examples

1. Enable site:

```
ee site enable example.com
```

2. Enable with dependency verification:

```
ee site enable example.com --verify
```

3. Force enable:

```
ee site enable example.com --force
```

Disable (stops and removes site containers)

```
ee site disable [<site-name>]
```

Example

```
ee site disable example.com
```

5) SSL verify/renew

Command

```
ee site ssl <site-name> [--force]
```

Options

1. `--force`: Force renewal.

Example

```
ee site ssl example.com --force
```

6) Site info **i**

Command

```
ee site info [<site-name>]
```

Example

```
ee site info example.com
```

7) List sites

Command

```
ee site list [--enabled] [--disabled] [--format=<format>]
```

Options

1. `--enabled`: Only enabled sites.
2. `--disabled`: Only disabled sites.
3. `--format=<format>`: Output format.
 1. `table` (default)
 2. `csv`
 3. `yaml`
 4. `json`
 5. `count`
 6. `text`

Examples

1. List all:

```
ee site list
```

2. List enabled:

```
ee site list --enabled
```

3. List disabled:

```
ee site list --disabled
```

4. JSON output:

```
ee site list --format=json
```

5. Count:

```
ee site list --format=count
```

8) Reload / Restart (pattern)

Your docs show the *same argument pattern* for several actions:

```
ee site reload --type=<type> <site-name> [--ssl=<value>] [--wildcard] [--skip-status-check]
ee site restart --type=<type> <site-name> [--ssl=<value>] [--wildcard] [--skip-status-check]
```

Notes

1. `--type=<type>`: Site type (`html`, `wp`, `php`, etc.).
2. `--ssl=<value>` and `--wildcard` follow the same meaning as create/update.
3. These commands are typically used for *re-provisioning / restarting* the site stack.

Example

```
ee site restart --type=html example.com
```

9) Share a site via ngrok

Command

```
ee site share <site-name> [--disable] [--refresh] [--token=<token>]
```

Options

1. `--disable`: Take the shared link down.
2. `--refresh`: Refresh if share link expired.
3. `--token=<token>`: ngrok auth token.

Examples

1. Share:

```
ee site share example.com
```

2. Refresh:

```
ee site share example.com --refresh
```

3. Disable:

```
ee site share example.com --disable
```

10) Clear cache (page/object)

Command

```
ee site clean [<site-name>] [--page] [--object]
```

Options

1. `--page`: Clear page cache.
2. `--object`: Clear object cache.

Examples

1. Clear both (no flags):

```
ee site clean example.com
```

2. Clear object cache:

```
ee site clean example.com --object
```

3. Clear page cache:

```
ee site clean example.com --page
```

WordPress Sites (`--type=wp`)

1) Create a WordPress site

Command

```
ee site create --type=wp <site-name> [--cache] [--vip] [--mu=<subdir>] [--mu=<subdom>] [--title=<title>] \  
  [--admin-user=<admin-user>] [--admin-pass=<admin-pass>] [--admin-email=<admin-email>] \  
  [--local-db] [--with-local-redis] [--php=<php-version>] \  
  [--dbname=<dbname>] [--dbuser=<dbuser>] [--dbpass=<dbpass>] [--dbhost=<dbhost>] \  
  [--dbprefix=<dbprefix>] [--dbcharset=<dbcharset>] [--dbcollate=<dbcollate>] \  
  [--skip-check] [--version=<version>] [--skip-content] [--skip-install] [--skip-status-check] \  
  \  
  [--ssl=<value>] [--wildcard] [--yes] [--force]
```

Most-used options

1. Core install

1. `--title=<title>`
2. `--admin-user=<admin-user>`
3. `--admin-pass=<admin-pass>`
4. `--admin-email=<admin-email>`

2. SSL

1. `--ssl=le` / `--ssl=self`
2. `--wildcard`

3. Caching / Redis

1. `--cache`: Enable Redis cache for WP.
2. `--with-local-redis`: Use a *site-local* Redis container.

4. Multisite

1. `--mu=subdir`: Multisite with subdirectories.
2. `--mu=subdom`: Multisite with subdomains.

5. Database placement

1. `--local-db`: Separate DB container for this site instead of global DB.
2. Remote DB:
 1. `--dbhost=<dbhost>`
 2. `--dbuser=<dbuser>`
 3. `--dbpass=<dbpass>`
 4. `--skip-check` (skip DB connection check)
 5. `--force` (reset remote DB if not empty)

6. WP version/content

1. `--version=<version>`: `latest`, `nightly`, or a version number.
2. `--skip-content`: Don't download default themes/plugins.
3. `--skip-install`: Skip `wp core install`.

PHP version

1. `--php=<php-version>` (as per your docs)
 1. `5.6`
 2. `7.2`
 3. `latest` (default)

Examples

1. Basic WP site:

```
ee site create example.com --type=wp
```

2. Multisite (subdir):

```
ee site create example.com --type=wp --mu=subdir
```

3. Multisite (subdom):

```
ee site create example.com --type=wp --mu=subdom
```

4. WP + Let's Encrypt:

```
ee site create example.com --type=wp --ssl=le
```

5. WP + wildcard SSL:

```
ee site create example.com --type=wp --ssl=le --wildcard
```

6. WP + self-signed SSL:

```
ee site create example.com --type=wp --ssl=self
```

7. WP with remote DB:

```
ee site create example.com --type=wp --dbhost=localhost --dbuser=username --dbpass=password
```

8. WP with custom title/admin (as in your docs; note: `--locale` appears in example though not listed in options):

```
ee site create example.com --type=wp --title=easyengine --locale=nl_NL \
  --admin-email=easyengine@example.com --admin-user=easyengine --admin-pass=easyengine
```

2) WP reload / restart (pattern)

These mirror the WP create flags:

```
ee site reload --type=wp <site-name> [same flags as create...]  
ee site restart --type=wp <site-name> [same flags as create...]
```

Example

```
ee site restart --type=wp example.com
```

Cron Management (`ee cron`)

1) Overview / help entry

```
ee cron
```

2) Create a cron job

Command

```
ee cron create [<site-name>] --command=<command> --schedule=<schedule> [--user=<user>]
```

Targets

1. Site cron: pass `<site-name>` (e.g., `example.com`)
2. Host cron: use `host` as site-name target (per your examples)

Schedule formats

1. Standard Linux cron: e.g. `* * * * *`
2. Shortcut macros:
 1. `@yearly` / `@annually` → `0 0 1 1 *`
 2. `@monthly` → `0 0 1 * *`
 3. `@weekly` → `0 0 * * 0`
 4. `@daily` / `@midnight` → `0 0 * * *`
 5. `@hourly` → `0 * * * *`
3. Fixed interval:
 1. `@every <duration>`
 1. duration units: `h`, `m`, `s`
 2. examples: `10m`, `1h`, `1h10m2s`

Examples

1. Run WP cron due events every 10 minutes (site):

```
ee cron create example.com --command='wp cron event run --due-now' --schedule='@every 10m'
```

2. Run every minute with Linux cron format:

```
ee cron create example.com --command='wp cron event run --due-now' --schedule='* * * * *'
```

3. Run every minute as `www-data`:

```
ee cron create example.com --command='wp cron event run --due-now' --schedule='* * * * *' --user=www-data
```

4. Add cron on the host machine:

```
ee cron create host --command='wp cron event run --due-now' --schedule='@every 10m'
```

5. Weekly cron on host:

```
ee cron create host --command='wp media regenerate --yes' --schedule='@weekly'
```

3) List cron jobs

Command

```
ee cron list [<site-name>] [--all]
```

Examples

1. List cron jobs (default view):

```
ee cron list
```

2. List cron jobs for one site:

```
ee cron list example.com
```

3. View all cron jobs:

```
ee cron list --all
```

4) Update a cron job

Command

```
ee cron update <id> [--site=<site>] [--command=<command>] [--schedule=<schedule>] [--user=<user>]
```

Examples

1. Change site target:

```
ee cron update 1 --site='example1.com'
```

2. Change command:

```
ee cron update 1 --command='wp cron event run --due-now'
```

3. Change command + user:

```
ee cron update 1 --command='wp cron event run --due-now' --user=root
```

4. Change schedule:

```
ee cron update 1 --schedule='@every 1m'
```

5) Run a cron job now ►

Command

```
ee cron run-now <cron-id>
```

Example

```
ee cron run-now 1
```

6) Delete a cron job ☐☐

Command

```
ee cron delete <cron-id>
```

Example

```
ee cron delete 1
```

Shell / Exec Into Containers (ee shell) ☐☐

Open a shell (interactive) or run a command (non-interactive)

Command

```
ee shell [<site-name>] [--user=<user>] [--service=<service>] [--command=<command>] [--skip-tty]
```

Options

1. `<site-name>`: Site name to enter.
2. `--user=<user>`: Exec as a specific user (e.g., `root`).
3. `--service=<service>`: Which service/container to exec into.
 1. default: `php`
4. `--command=<command>`: Run a single command non-interactively.
5. `--skip-tty`: Don't allocate a TTY (useful in CI).

Examples

1. Open default shell (PHP service):

```
ee shell example.com
```

2. Open shell as root:

```
ee shell example.com --user=root
```

3. Open shell into nginx:

```
ee shell example.com --service=nginx
```

4. Run a one-shot command in nginx:

```
ee shell example.com --service=nginx --command='nginx -t && nginx -s reload'
```

“Fast Lookup” Index □

1. **Create HTML site:** `ee site create --type=html example.com`
2. **Create WP site:** `ee site create --type=wp example.com`
3. **Add SSL:** `ee site update example.com --ssl=le`
4. **Force renew SSL:** `ee site ssl example.com --force`
5. **Enable / Disable:** `ee site enable example.com` / `ee site disable example.com`
6. **Info:** `ee site info example.com`
7. **List sites:** `ee site list --format=json`
8. **Share via ngrok:** `ee site share example.com`
9. **Clear cache:** `ee site clean example.com --page` (or `--object`)
10. **Cron add:** `ee cron create example.com --command='...' --schedule='@every 10m'`
11. **Cron list:** `ee cron list --all`

12. **Shell:** `ee shell example.com --service=php`

If you tell me which EasyEngine major version you're on (v4/v5+) and which site types you actually use (html/php/wp), I can tailor this into a *one-page* condensed cheat sheet plus a *troubleshooting mini-section* (common errors + exact commands to diagnose).

Revision #1

Created 2026-04-15 16:40:26 UTC by art10m

Updated 2026-04-15 16:43:02 UTC by art10m