The setup wizard on shared hosting

How customers without shell access install Exponential in the browser with the setup wizard, and what the host prepares: code upload, writable directories, the .htaccess file, HTTPS, the database, the maintenance hold…

Source file: `doc/eng-us/help/hosting/provisioning/setup-wizard-on-shared-hosting.md` Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/help/hosting/provisioning/setup-wizard-on-shared-hosting.md

This page is for hosts whose customers install Exponential themselves in a browser, typically on shared hosting without shell access, and for the support staff who help them. It answers: what the host must prepare, what the customer sees, and how the wizard protects the site while it runs. The page-by-page reference is Chapter 5.

> **Warning**
>
> `.htaccess` **slows down every request.** When `.htaccess` files are enabled (`AllowOverride` anything but `None`), let alone used, Apache looks for them and reads and evaluates them on every request, in every directory of the path, and page loading times go up drastically (measured). The only real fix is to switch the feature off for the virtual host: `AllowOverride None` in the HTTPS virtual host, and the rules in the virtual host's own configuration as mod_rewrite rules. That is far faster in production speed tests and well worth the time it takes to optimize a production configuration. Switch over right away: copy the rules of the example HTTPS virtual host into your virtual host (with the static cache, also `staticcache_offload.conf`), set `AllowOverride None`, run `apachectl configtest`, reload Apache and test the site. The same rules for other servers: nginx, LiteSpeed, Caddy, IIS, Velocity (built in); where virtual host rules go in a control panel: Plesk, cPanel, DirectAdmin, ISPConfig. **On LiteSpeed or OpenLiteSpeed,** the server's own page cache can also answer anonymous visitors and crawlers without PHP: Page caching with LiteSpeed and OpenLiteSpeed.

Contents

1. When the wizard is the right choice
2. What the host prepares
3. What the customer does
4. While it runs
5. After the wizard
6. Pre-filling pages with kickstart.ini

When the wizard is the right choice

The installation book recommends it for "shared hosting without shell access … it needs only a browser and FTP access" (4.5). It is also "the only method that shows each decision on a page with an explanation". When you can run commands for the customer, exp:install is faster and

scriptable.

What the host prepares

Item	Why
PHP 8.0 or later (PHP 7.4 usable without support) with the required extensions; Apache with `mod_rewrite` and PHP-FPM is "the classic setup; shared hosting"	Requirements at a glance
The code, with `vendor/` (the dependencies), in the customer's web space	the wizard cannot run Composer; upload a complete release
`settings/`, `var/` and `design/` writable by the user PHP runs as	the System check tests `[directory_permissions] CheckList` (Writable directories and permissions)
`.htaccess` in place: a copy of the shipped `.htaccess_root`	the rewrite rules that keep internal files from being served
HTTPS on the domain before the wizard is opened	"The database password and the administrator password travel in those hidden fields with every page" (5.1)
A database and its user, or nothing for SQLite 3	SQLite needs only a writable `var/storage/sqlite3/`

> **Tip**
> If `.htaccess` is missing, the wizard's **Site security** page asks for it with "cp .htaccess_root .htaccess" and "If you do not have shell access, you will have to copy the file using an FTP client or ask your hosting provider to do this for you" (5.3.13). Put it there in advance and the page is skipped.

The host's provisioning can do that with the console, before the customer opens the wizard. As root, the copy needs `--allow-root-user` and should be given to the account's user:

Shell

```
cd /path/to/the/installation
php bin/php/console exp:setup:htaccess --apply --allow-root-user --owner=<account-user>:<account-group>
php bin/php/console exp:setup:config --apply --allow-root-user --owner=<account-user>:<account-group>   # optional
```

Without `--owner`, a new file gets the owner of the template it is copied from, never root. Details: Setting up .htaccess and config.php from the console.

What the customer does

1. Opens the site's address. While `[SiteAccessSettings] CheckValidity=true` (the shipped value), "whatever path you open, the wizard's first page appears" (5.1).
2. Walks the pages: welcome, system check, mail, database, languages, site package, siteaccesses, site details, administrator, and the installation itself (5.3).
3. Chooses an administrator password. The wizard "refuses a short or well-known one" (The default administrator password).

4. Reaches the Finished page with the addresses of the new site.

If the System check fails, it prints the cause and the fix, for permissions the exact `chmod` and `chown` commands; on shared hosting those are often something only the host can run.

While it runs

The wizard holds the site for the browser that started it (5.2):

- The first posted step writes `var/maintenance.json` with a 15-minute lease, renewed by every further step.
- Every other visitor gets HTTP 503 "The site is being set up".
- The Finished page ends the hold; an abandoned wizard "does not keep the site offline: once the lease has run out, the marker no longer counts".

After the wizard

- `CreateSites` writes `CheckValidity=false` to `settings/override/site.ini.append.php`; "from then on the site is served normally and the wizard can no longer be reached".
- On an installed site, `setup/init` only shows "This site is already installed" and writes nothing (5.2).
- Install the crontab for the customer if your panel does not let them (Cron and shell access).
- Go through the go-live checklist.

Pre-filling pages with kickstart.ini

The wizard reads `kickstart.ini` in the installation root: "When the wizard finds a section for a step, it pre-fills that page, or skips it when the section says `Continue=true`" (4.1). A host can use this to fill in the database page with the database it created, so the customer never sees the database password. Remove the file after the install; it holds passwords in clear text.

A public demo through the wizard. The setup wizard reads `EXP_DEMO_MODE=1` from the web server's PHP (for example `SetEnv` in `.htaccess`); its **Site administrator** page then shows a **Demo mode** note and accepts the published password, also `publish`. Demo mode for hosts.

Related pages

- One-click installers
- Shared hosting
- Apache with PHP-FPM
- Setting up .htaccess and config.php from the console
- Free documentation: this page is free to copy, translate and improve under the GNU FDL 1.3.

> **Become an Exponential Partner**
>
> Partners share their information, provide funding, contribute back and develop the products with us, and that is how Exponential grows. A partner website and dashboard are

> planned for mid-2027: please become a partner today to speed up our growth rate and land new features faster.
>
> Partners: start here | How to apply

Exponential Copyright © 1998-2026 Exponential Foundation and 7x
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".
This documentation is free documentation, licensed under the GNU Free Documentation License 1.3 or later:
https://www.gnu.org/licenses/fdl-1.3.html.