

What Is Self-Hosting?

“ **The act of providing or serving digital content or an online service typically delivered by a business.**

The service or content is generally served locally from your own hardware. Often "self-hosters" use older Enterprise-grade hardware from their home internet connections however they also use other hosting providers hardware. This is still considered self-hosting.

One of the easiest things to self-host with the lowest barrier to entry is a website. For the most basic website of your own, all you need is a domain name and a webserver. Then you throw a few lines of HTML in a file and you have yourself a "website". With a service like Let's Encrypt, securing the site with a SSL certificate is easy too.

A lot of different services that you can self-host are "websites". There are dynamic sites with robust content management systems like Joomla!, Drupal, WordPress, or b2evolution. There are forums like phpBB, MyBB, vBulletin, Discourse, etc. Knowledge bases like DokuWiki, MediaWiki, BookStack, or Gollum are also websites. These websites only require a webserver, an interpreter (PHP), and a database (SQLite, PostgreSQL, MySQL).

Just about everything these days has a web UI or frontend to make things easier. HTTP/HTML/JS are well-understood standards that are ubiquitous. There are many libraries for converting or presenting your content in a web-friendly way for almost all programming languages you can learn.

It can be hard for someone unfamiliar to find the difference between the "website" frontend and the content backend. Sometimes the difference is almost non-existent. Sometimes there are many layers and systems working behind the scenes to make it happen.

It may be better to say that everything can be "accessed" through a website, even if it isn't one *per-se*. And if it can't, there's probably a separate piece of software that makes a web UI for it.

Examples of services with a web UI or separate web-based frontends are: BitTorrent clients like qBittorrent/Transmission, media streaming servers like Jellyfin/Navidrome, file synchronization services like Nextcloud/ownCloud/Seafile, communication services like Synapse/InspIRCd/jabberd/Mumble, and many more.

Other services use the server-client model where the entire package is in two parts. The server part that runs at all time to serve content and the client part that connects to have content served to it. Examples are: game servers like Rust/Minecraft/Factorio, FTP servers, email servers, and more.

Revision #1

Created 24 June 2025 19:03:43 by Nicolas

Updated 24 June 2025 19:04:06 by Nicolas