

Preparing PHP



- [PHP Installation](#)
 - [Required PHP Modules](#)
 - [Required PHP Database Connectors](#)
 - [Recommended General PHP Modules](#)
 - [Recommended PHP Caching Modules](#)
 - [Recommended PHP CLI Modules](#)
 - [PHP Modules for Media Management](#)
 - [PHP Modules for Specific Applications](#)
 - [PHP ini Settings](#)
 - [Notes on PHP ini Configuration](#)
 - [PHP Module Quick Reference Table](#)
 - [Further Resources](#)
-

Before installing Nextcloud Server, ensure your PHP environment is properly configured. This includes installing the correct PHP version, enabling required PHP modules, and adjusting important php.ini settings. This guide explains which PHP modules are necessary, which are recommended for optimal performance and compatibility, and how to configure your PHP environment for both web server and command-line usage.



You can safely ignore this chapter if you plan to use a turnkey Nextcloud Server installation method (such as AIO, Snap, NCP, or Community Docker). Those installation methods provide PHP environments that are already pre-configured for use with Nextcloud Server. For guidance regarding customizing PHP in those environments, refer to the documentation provided specifically for or by those install methods.

PHP Installation

Refer to your OS distribution's documentation for instructions for establishing a base PHP installation. It may be possible to choose among several versions of PHP. Refer to [System requirements](#) to see which versions of PHP are supported by this release of Nextcloud Server. After completing a base PHP installation, follow the below guidance to configure your new PHP installation for your new Nextcloud Server deployment.

Required PHP Modules

The following PHP modules **must** be installed and enabled for Nextcloud Server to function:

- ctype (included with PHP)
- curl
- DOM
- fileinfo (included with PHP)
- filter (only on Mageia and FreeBSD)
- GD
- xml (provides SimpleXML, XMLReader and XMLWriter; requires Linux package libxml2 version $\geq 2.7.0$)
- mbstring
- OpenSSL (included with PHP)
- posix
- session (included with PHP)
- zip
- zlib



The PHP "xml" extension is commonly packaged as php-xml or shown as libxml by OS package managers. This extension provides the underlying libxml2 bindings and exposes SimpleXML, XMLReader and XMLWriter. Ensure the corresponding php-xml (or distribution-specific) package is installed so that SimpleXML, XMLReader and XMLWriter are available to PHP.

The ctype, fileinfo, and OpenSSL modules are generally included and enabled in PHP by default. Often some of the other required modules are automatically installed by OS distribution package managers.



PHP Version-Specific Information:

- **PHP 8.3 and 8.4:** Several modules are now bundled with PHP by default, such as curl, zlib, and others. When upgrading to these versions, some modules that were previously optional may already be enabled. If you encounter "module not found" errors when running standard package installation commands, the module may already be bundled and enabled — verify this by using the module check command below.
- **Debian/Ubuntu packages:** Module availability varies by distribution version. If a listed module cannot be found in your package manager, check the PHP documentation or your distribution's PHP module list to confirm if the feature is bundled with your PHP version.

How to check if a module is enabled:

- Run `php -m | grep -i <module_name>`. If you see output, the module is active.

“ ”

The filter module is required only on Mageia and FreeBSD.

Required PHP Database Connectors

Install the PHP connector module for the database you plan to use (choose one):

- pdo_sqlite (>= 3, usually not recommended for performance reasons)
- pdo_mysql (MySQL/MariaDB)
- pdo_pgsql (PostgreSQL)

Recommended General PHP Modules

These modules are not required, but are highly recommended to improve functionality or security:

- ``intl``: Fixes sorting of non-ASCII characters and improves language translation performance.
- ``sodium``: Provides Argon2 password hashing (needed if using PHP < 8.4 and PHP was built without libargon2). Starting with PHP 8.0, sodium is usually enabled by default. In PHP 8.4, sodium is typically bundled.

“bcrypt will be used if Argon2 is unavailable, but if passwords were previously hashed with Argon2 (such as when migrating an existing Nextcloud Server installation to a new server environment) and this module is missing, accounts will not be able to log-in).

- ``sysvsem``: Enables System V semaphores used by Nextcloud to coordinate preview generation across PHP processes. Recommended; if missing, previews still work but may be less reliable under heavy load.

Recommended PHP Caching Modules

Memory caching is not required so these modules are not required, but are highly recommended for optimal performance and reliability. Choose and install your preferred combination of memory caching modules:

- APCu (>= 4.0.6)
- redis / phredis (>= 2.2.6, required for Transactional File Locking)
- memcached (an older alternative to redis that is not recommended for new installations)

“ ”

Memory caching is highly recommended for optimal performance. In most cases, a combination of APCu and redis are the best choice for new installations. You can also use the `KeyValueCache` cache option which provides Redis-like functionality without requiring installing a PHP module, see [KeyValueCache configuration in Nextcloud \(config.php\)](#).

See [Memory caching](#) for configuration details.

Recommended PHP CLI Modules

For command-line processing (optional):

- ``pcntl``: Allows command interruption (e.g., via `ctrl-c`).

“ Ensure `pcntl_signal` and `pcntl_signal_dispatch` are *not* disabled in your `php.ini` by the `disable_functions` option.

For command-line updater (optional):

- ``phar``: Required to run the updater with:

“ `sudo -E -u www-data php /var/www/nextcloud/updater/updater.phar`

PHP Modules for Media Management

Image meta data and orientation (optional):

- ``exif``: Image meta data loading and rotation

Preview Generation (optional):

- `imagemagick` (for image previews)
- `avconv` or `ffmpeg` (for video previews)
- `OpenOffice` or `LibreOffice` (for document previews)

“ □

If previewing PDF files fails with a "not authorized" error, you may need to adjust the `imagemagick` policy file. See <https://cromwell-intl.com/open-source/pdf-not-authorized.html>

See [Previews configuration](#) for additional preview generation context.

PHP Modules for Specific Applications

Some optional Nextcloud apps/functionality require additional modules. Install as needed:

- ``ldap``: LDAP integration
- ``smbclient``: SMB/CIFS integration (see [SMB/CIFS](#))
- ``ftp``: FTP storage or external user authentication
- ``imap``: External user authentication

Recommended/Optional:

- ``gmp``: SFTP storage

PHP ini Settings

Adjust the following settings in your `php.ini` as needed for Nextcloud:

- `disable_functions`: Avoid disabling functions unless necessary.
- `max_execution_time`: See [Uploading big files > 512MB](#)
- `memory_limit`: Should be at least 512MB. See also [Uploading big files > 512MB](#)
- `opcache.enable` and related settings: See [Memory caching](#) and [Server tuning](#)
- `open_basedir`: See [Hardening and security guidance](#)
- `upload_tmp_dir`: See [Uploading big files > 512MB](#)

Notes on PHP ini Configuration

- **Multiple `php.ini` files:**

- You may need to configure settings in more than one `php.ini` file (e.g., for web server and CLI).

- Web server:
/etc/php/<version>/apache2/php.ini
or
/etc/php/<version>/fpm/php.ini
- CLI (used by Nextcloud CRON jobs):
/etc/php/<version>/cli/php.ini

- **Find which php.ini is active for each SAPI:**

“ ◦ Use `php --ini` for CLI, or check `phpinfo()` in a web page.

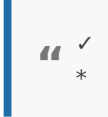
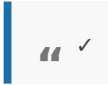
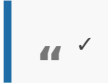
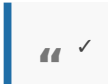
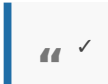
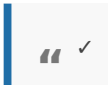
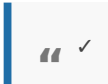
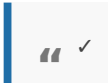
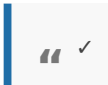
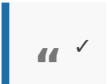
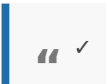
- **Search for a parameter:**

“ ◦ Run `grep -r <parameter_name> /etc/php` (e.g., `grep -r date.timezone /etc/php`)

- **Replace `<version>` with your actual PHP version (e.g., 8.1, 8.2, etc.).**

PHP Module Quick Reference Table

Module	Required	Recommended	For Specific App	Description
ctype	“ ✓			Core functionality
curl	“ ✓			HTTP requests
DOM	“ ✓			Document Object Model (XML/HTML handling)
fileinfo	“ ✓			File type detection

Module	Required	Recommended	For Specific App	Description
filter*				Data filtering and validation (Mageia/FreeBSD)
GD				Image processing
xml				XML parsing (libxml2 >= 2.7.0) — provides the php 'xml' extension which exposes SimpleXML, XMLReader and XMLWriter
mbstring				Multibyte character handling
OpenSSL				Secure communications
posix				POSIX functions
session				Session support
zip				Zip file handling
zlib				Compression and decompression
intl				Improves translations and sorting
sodium				Argon2 password hashing

Module	Required	Recommended	For Specific App	Description
sysvsem		“ ✓		System V semaphore support used to coordinate preview generation across PHP processes.
ldap			“ ✓	LDAP integration
smbclient			“ ✓	SMB/CIFS integration
ftp			“ ✓	FTP storage/authentication
imap			“ ✓	External user authentication
gmp			“ ✓ (o p t i o n a l)	SFTP storage
exif			“ ✓ (o p t i o n a l)	Image rotation in Pictures app
apcu		“ ✓		Performance caching

Module	Required	Recommended	For Specific App	Description
memcached		“✓		Performance caching
redis		“✓		Transactional File Locking
imagick			“✓ (o p t i o n a l)	Image previews
avconv/ffmpeg			“✓ (o p t i o n a l)	Video previews
Open/LibreOffice			“✓ (o p t i o n a l)	Document previews

Module	Required	Recommended	For Specific App	Description
pcntl			“ ✓ (o p t i o n a l) ”	Command interruption in CLI
phar			“ ✓ (o p t i o n a l) ”	Needed for command-line updater

*The filter module is required only on Mageia and FreeBSD.

Further Resources

- For more details on each module, consult the [official PHP documentation](#).
- Refer to your OS distribution's documentation for the specifics of installing PHP modules in your environment.
- The words *extension* and *module* are interchangeable within PHP. We use the word *modules* in our documentation.
- Always restart your web server and PHP-FPM after making changes to an php.ini file or installed modules.