

General troubleshooting



- [Bugs](#)
 - [General troubleshooting](#)
 - [Disable 3rdparty / non-shipped apps](#)
 - [Internal Server Errors](#)
 - [Nextcloud log files](#)
 - [PHP version and information](#)
 - [Debugging sync issues](#)
 - [Common problems / error messages](#)
 - [Troubleshooting Web server and PHP problems](#)
 - [Logfiles](#)
 - [Web server and PHP modules](#)
 - [Troubleshooting WebDAV](#)
 - [Service discovery](#)
 - [Troubleshooting sharing](#)
 - [Users' Federated Cloud IDs not updated after a domain name change](#)
 - [Troubleshooting contacts & calendar](#)
 - [Troubleshooting data-directory](#)
 - [Moving the data directory / changing the datadirectory path](#)
 - [Troubleshooting quota or size issues](#)
 - [Troubleshooting encrypted files](#)
 - [Fair Use Policy](#)
 - [Other issues](#)
-

If you have trouble installing, configuring or maintaining Nextcloud, please refer to our community support channels:

- [The Nextcloud Forums](#)

The Nextcloud forums have a [FAQ page](#) where each topic corresponds to typical mistakes or frequently occurring issues

Please understand that all these channels essentially consist of users like you helping each other out. Consider helping others out where you can, to contribute back for the help you get. This is the only way to keep a community like Nextcloud healthy and sustainable!

If you are using Nextcloud in a business or otherwise large scale deployment, note that Nextcloud GmbH offers commercial support options.

Bugs

If you think you have found a bug in Nextcloud, please:

- Search for a solution (see the options above)
- Double-check your configuration

If you can't find a solution, please use our [bugtracker](#). You can generate a configuration report with the [occ config command](#), with passwords automatically obscured.

General troubleshooting

Check the Nextcloud [System requirements](#), especially supported browser versions.

When you see warnings about `code integrity`, refer to [Code signing](#).

Disable 3rdparty / non-shipped apps

It might be possible that 3rd party / non-shipped apps are causing various different issues. Always disable 3rd party apps before upgrades, and for troubleshooting. Please refer to the [Apps commands](#) on how to disable an app from command line.

Internal Server Errors

An Internal Server Error, sometimes called a "500 error", indicates that the web server encountered an unexpected condition that prevented it from fulfilling the request.

This error response is a generic "catch-all" response. To find out the source of the error you will need to check your Nextcloud log (located in `data/nextcloud.log` by default) and possibly your web server's error log (depending on where the failure is occurring).



Whenever possible, Nextcloud will include the "Request id" in the error. This request ID can be searched for in your Nextcloud log file to find entries associated with the failing transaction.

Nextcloud log files

The Nextcloud log file is located in the data directory by default - e.g. `data/nextcloud.log`. If the Web UI is still reachable, it is also available via *Administration settings->Logging*.



When asking for help, the entire raw log entry is generally required.

For some situations you may need to adjust the log level in your `config.php` file. Please see [Logging](#) for more information on these log levels.

Some logging - for example JavaScript console logging - needs debugging enabled. Edit `config/config.php` and change `'debug' => false,` to `'debug' => true,` Be sure to change it back when you are finished.

For JavaScript issues you will also need to view the javascript console. All major browsers have developer tools for viewing the console, and you usually access them by pressing F12.

PHP version and information

You will need to know the PHP version and configuration that is in-use on your Nextcloud server. This will not necessarily be the same version and configuration as can be reached from the command-line. The simplest way to gather this information is by using what's commonly referenced as `phpinfo()`.

The most accurate - and easiest - way to access `phpinfo` is by checking it from within Nextcloud itself. Of course, this requires that Nextcloud is functioning enough that you can log in as an administrator and access the **Administration settings -> System** menu. If so, you can enable the exposure of `phpinfo` data by toggling it on via `occ`:

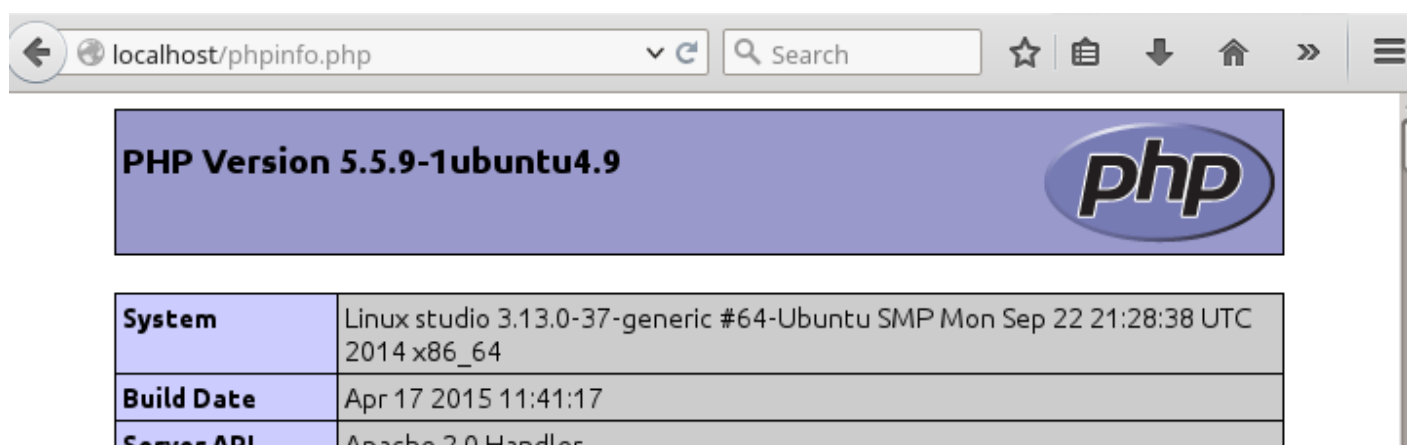
```
./occ config:app:set --value=yes serverinfo phpinfo
```

From then on a new button labeled **Show phpinfo** will be visible in the web interface under **Administration settings -> System**. Clicking it will expose just about everything you may want to know about your PHP environment.

If accessing the Nextcloud web interface is not an option, you may create a plain-text file named **phpinfo.php** and place it in your Web root, for example `/var/www/html/phpinfo.php`. (Your Web root may be in a different location; your Linux distribution documentation will tell you where.) This file contains just this line:

```
<?php phpinfo(); ?>
```

Open this file in a Web browser by pointing your browser to `localhost/phpinfo.php`:



Your PHP version is at the top, and the rest of the page contains abundant system information such as active modules, active `.ini` files, and much more. When you are finished reviewing your information you must delete `phpinfo.php`, or move it outside of your Web directory, because it is a security risk to expose such sensitive data.

Debugging sync issues

“ ”

The data directory on the server is exclusive to Nextcloud and must not be modified manually.

Disregarding this can lead to unwanted behaviors like:

- Problems with sync clients
- Undetected changes due to caching in the database

If you need to directly upload files from the same server please use a WebDAV command line client like `cadaver` to upload files to the WebDAV interface at:

Common problems / error messages

Some common problems / error messages found in your logfiles as described above:

- `SQLSTATE[HY000] [1040] Too many connections` -> You need to increase the connection limit of your database, please refer to the manual of your database for more information.
- `SQLSTATE[HY000]: General error: 5 database is locked` -> You're using `SQLite` which can't handle a lot of parallel requests. Please consider converting to another database like described in [Converting database type](#).
- `SQLSTATE[HY000]: General error: 2006 MySQL server has gone away` -> Please refer to [Troubleshooting](#) for more information.
- `SQLSTATE[HY000] [2002] No such file or directory` -> There is a problem accessing your SQLite database file in your data directory (`data/nextcloud.db`). Please check the permissions of this folder/file or if it exists at all. If you're using MySQL please start your database.
- `Connection closed / Operation cancelled` -> This could be caused by wrong `KeepAlive` settings within your Apache config. Make sure that `KeepAlive` is set to `On` and also try to raise the limits of `KeepAliveTimeout` and `MaxKeepAliveRequests`.
- `No basic authentication headers were found` -> This error is shown in your `data/nextcloud.log` file. Some Apache modules like `mod_fastcgi`, `mod_fcgid` or `mod_proxy_fcgi` are not passing the needed authentication headers to PHP and so the login to Nextcloud via WebDAV, CalDAV and CardDAV clients is failing.

Troubleshooting Web server and PHP problems

Logfiles

When having issues the first step is to check the logfiles provided by PHP, the Web server and Nextcloud itself.

“ ”

In the following the paths to the logfiles of a default Debian installation running Apache2 with `mod_php` is assumed. On other Web servers, Linux distros or

operating systems they can differ.

- The logfile of Apache2 is located in `/var/log/apache2/error.log`.
- The logfile of PHP can be configured in your `/etc/php/8.3/apache2/php.ini`. You need to set the directive `log_errors` to `On` and choose the path to store the logfile in the `error_log` directive. After those changes you need to restart your Web server.
- The logfile of Nextcloud is located in the data directory `/var/www/nextcloud/data/nextcloud.log`.

Web server and PHP modules



Lighttpd is not supported with Nextcloud, and some Nextcloud features may not work at all on Lighttpd.

There are some Web server or PHP modules which are known to cause various problems like broken uploads/downloads. The following shows a draft overview of these modules:

1. Apache

- `mod_pagespeed`
- `mod_evasive`
- `mod_security`
- `mod_reqtimeout`
- `mod_deflate`
- `mod_spdy`
- `mod_dav`
- `mod_xsendfile` / X-Sendfile (causing broken downloads if not configured correctly)

2. NginX

- `ngx_pagespeed`
- `HttpDavModule`
- X-Sendfile (causing broken downloads if not configured correctly)

3. PHP

- Tideways
- eAccelerator

Troubleshooting WebDAV

Nextcloud uses SabreDAV, and the SabreDAV documentation is comprehensive and helpful.

See:

- [SabreDAV FAQ](#)
- [Web servers](#) (Lists lighttpd as not recommended)
- [Working with large files](#) (Shows a PHP bug in older SabreDAV versions and information for mod_security problems)
- [0 byte files](#) (Reasons for empty files on the server)
- [Clients](#) (A comprehensive list of WebDAV clients, and possible problems with each one)
- [Finder, OS X's built-in WebDAV client](#) (Describes problems with Finder on various Web servers)

There is also a well maintained FAQ thread available at the [ownCloud Forums](#) which contains various additional information about WebDAV problems.

Service discovery

Some clients - especially on iOS/macOS - have problems finding the proper sync URL, even when explicitly configured to use it.

If you want to use CalDAV or CardDAV clients or other clients that require service discovery together with Nextcloud it is important to have a correct working setup of the following URLs:

```
https://example.com/.well-known/carddav
```

```
https://example.com/.well-known/caldav
```

Those need to be redirecting your clients to the correct endpoints. If Nextcloud is running at the `DocumentRoot` of your Web server the correct URL is `https://example.com/remote.php/dav` for CardDAV and CalDAV and if running in a subfolder like `nextcloud`, then the correct URL is `https://example.com/nextcloud/remote.php/dav`.



On Debian/Ubuntu systems, the Apache `DocumentRoot` defaults to `/var/www/html`.
On other distributions, this path may differ. Adjust paths in the examples below

accordingly.

For the first case, the `.htaccess` file shipped with Nextcloud should do this work for you when you're running Apache. You need to make sure that your Web server is using this file. Additionally, you need the `mod_rewrite` Apache module installed and `AllowOverride All` set in your `apache2.conf` or `vHost-file` to process these redirects. When running Nginx please refer to [NGINX configuration](#).

For the second case, you need to add the following to `/etc/apache2/apache2.conf`, replacing `/var/www/html` with your actual `DocumentRoot` if it differs:

```
<Directory /var/www/html>
    AllowOverride FileInfo
</Directory>
```

`AllowOverride FileInfo` is sufficient for the rewrite directives used by service discovery. If you already have `AllowOverride All` set for this directory, no change is needed. Next, create or edit the `.htaccess` file within the `DocumentRoot` of your Web server and add the following lines:

```
<IfModule mod_rewrite.c>
    RewriteEngine on
    RewriteRule ^\.well-known/carddav /nextcloud/remote.php/dav [R=301,L]
    RewriteRule ^\.well-known/caldav /nextcloud/remote.php/dav [R=301,L]
    RewriteRule ^\.well-known/webfinger /nextcloud/index.php/.well-known/webfinger [R=301,L]
    RewriteRule ^\.well-known/nodeinfo /nextcloud/index.php/.well-known/nodeinfo [R=301,L]
</IfModule>
```

Make sure to change `/nextcloud` to the actual subfolder your Nextcloud instance is running in.



If you put the above directives directly into an Apache configuration file (usually within `/etc/apache2/`) instead of `.htaccess`, you need to prepend the first argument of each `RewriteRule` option with a forward slash `/`, for example `^/\.well-known/carddav`. This is because Apache normalizes paths for the use in `.htaccess` files by dropping any number of leading slashes, but it does not do so for the use in its main configuration files.

If you are running NGINX, make sure `location = /.well-known/carddav {` and `location = /.well-known/caldav {` are properly configured as described in [NGINX configuration](#), adapt to use a subfolder if necessary.

Now change the URL in the client settings to just use:

```
https://example.com
```

instead of e.g.

```
https://example.com/nextcloud/remote.php/dav/principals/username .
```

There are also several techniques to remedy this, which are described extensively at the [Sabre DAV website](#).

Troubleshooting sharing

Users' Federated Cloud IDs not updated after a domain name change

1. run Database query

```
DELETE FROM oc_cards_properties WHERE name = 'CLOUD' AND addressbookid = (select id from oc_addressbooks where principaluri = 'principals/system/system' AND uri = 'system');
```

2. run occ commands

```
occ dav:sync-system-addressbook
occ federation:sync-addressbooks
```

Troubleshooting contacts & calendar

“ ”

Please also refer to the troubleshooting article in the groupware section: [Troubleshooting](#).

Troubleshooting data-directory

Moving the data directory / changing the `datadirectory` path

For local storage, Nextcloud identifies storage by its absolute path on disk. Ideally, the location of the data directory should not change after deployment. If this is a new installation, consider reinstalling with your preferred directory location before moving into production.



If you must change the `datadirectory` path -- unless the transition is handled carefully -- Nextcloud will treat its content as "new storage." This can trigger duplicate or orphaned files, lost file metadata, and the loss of previously shared links.

For safely moving the data directory, the recommended actions are:

1. Make sure no cron jobs are running and, if using system cron, that the Nextcloud crontab entry is disabled.
2. Stop web/app server(s).
3. Move `/data` to the new location (ensure you also move hidden/dot files such as `.ncdata`).
4. Create a symlink from the original location to the new location.
5. Ensure permissions are still correct (including for any parent folders).
6. Restart web/app server(s).
7. Re-enable Nextcloud's system crontab entry (if applicable).



You may need to configure your web server to support symlinks.

It is also possible to move the data directory without using symlinks, but this requires manually modifying the internal `oc_storages` database table:

1. Make sure no cron jobs are running and, if using system cron, that the Nextcloud crontab entry is disabled.
2. Stop web/app server(s).
3. Move `/data` to the new location (ensure you also move hidden/dot files such as `.ncdata`).

4. Update the value of `datadirectory` in your `config.php`.
5. Edit the database: In the `oc_storages` table, update the path portion of the `id` field of the entry beginning with `local::/old-data-dir/` (e.g., change `local::/old-data-dir/` to `local::/new-data-dir/`).
6. Ensure permissions are still correct (including for any parent folders).
7. Restart web/app server(s).
8. Re-enable Nextcloud's system crontab entry (if applicable).



This method is not supported and you risk breaking your database. Always make sure you have up-to-date backups -- including your database -- and a working (tested) restore process **before** attempting this.

Troubleshooting quota or size issues

Sometimes it can happen that the used space reported in the web UI or with `occ user:info $userId` does not match the actual data stored in the user's `data/$userId/files` directory.



Metadata, versions, trashbin and encryption keys are not counted in the used space above. Please refer to the [quota documentation](#) for details.

Running the following command can help fix the sizes and quota for a given user:

```
sudo -E -u www-data php occ files:scan -vvv <user-id>
```

If **encryption was enabled earlier on the instance and disabled later on**, it is likely that some size values in the database did not correctly get reset upon decrypting. You can run the following SQL query to reset those after **backing up the database**:

```
UPDATE oc_filecache SET unencrypted_size=0 WHERE encrypted=0;
```

Troubleshooting encrypted files



Please also refer to the troubleshooting section in the encryption chapter:
[Server-side Encryption](#).

Fair Use Policy

Nextcloud is open source and you can host it for free on your own server or at a provider.

Nextcloud recommends Using Nextcloud Enterprise for deploying instances with more than 500 users. With that size, issues like a broken server or a data leak become very serious.

If there is an issue with the server, 500 people can't work. A data leak would risk the data of many users. In short, the server should be considered mission-critical. We believe you and your users would have a better experience with Nextcloud Enterprise.

Nextcloud Enterprise is pre-configured and optimised for the needs of professional organisations rather than home users. It comes with support, security and scaling benefits, compliance expertise, and access to our knowledge about running a successful Nextcloud, to get the best possible experience for users and admins. This also reduces the load on our home user forum

<http://help.nextcloud.com> from issues unique to big deployments.

Nextcloud provides some infrastructure components needed for Nextcloud servers to run reliably. This includes notification, our app store and more. To ensure these resources do not get overloaded by administrators who run Nextcloud for thousands of users without providing financial resources to Nextcloud in return, these components are limited and will not work for more than 500 users.

We believe all organisations who run Nextcloud for hundreds of users should be officially supported. We know there can be financial restrictions for non-profit organisations and, as we want everybody to have a chance to get the most out of Nextcloud, we have special offers for NGOs, small schools and other non-profits. Please reach out to talk to us about what is possible through the [contact form on our site](#) or ask your system administrator to reach out.

Other issues

Some services like *Cloudflare* can cause issues by minimizing JavaScript and loading it only when needed. When having issues like a not working login button or creating new users make sure to disable such services first.

- 📄 #3
- 📄 cocobook-sync 📄 2026-10-28 16:30:51 UTC
- 📄 cocobook-sync 📄 2026-10-29 01:09:27 UTC