# Using the occ command

□□


- Running occ
  - Environment variables
- Enabling autocompletion
- Limitations in maintenance mode
- Debugging
- Command reference


Nextcloud's `occ` command (origins from "ownCloud Console") is Nextcloud's command-line interface. You can perform many common server operations with `occ`, such as installing and upgrading Nextcloud, manage users, encryption, passwords, LDAP setting, and more.

`occ` is in the `nextcloud/` directory; for example `/var/www/nextcloud` on Ubuntu Linux. `occ` is a PHP script. **You must run it as your HTTP user** to ensure that the correct permissions are maintained on your Nextcloud files and directories.

# Running occ

You must run `occ` as your HTTP user so that the file ownership and permissions on your Nextcloud data directory stay consistent with the web server.

The HTTP user is different on the various Linux distributions:

- The HTTP user and group in Debian/Ubuntu is www-data.
- The HTTP user and group in Fedora/CentOS is apache.
- The HTTP user and group in Arch Linux is http.
- The HTTP user in openSUSE is wwwrun, and the HTTP group is www.

> “□□
>
> APCu is disabled by default for the command-line mode of PHP, which can cause issues with Nextcloud's `occ` command. Please make sure you set the

> `apc.enable_cli` parameter to `1` in your PHP CLI's `php.ini` configuration file or append `--define apc.enable_cli=1` each time you invoke `occ` - e.g.:
>
> ```
> sudo -u www-data php --define apc.enable_cli=1 occ config:list system
> ```
>
> If you fail to do this, you will receive output such as the following:
>
> ```
> An unhandled exception has been thrown:
> OCP\HintException: [0]: Memcache \OC\Memcache\APCu not available for local
> cache (Is the matching PHP module installed and enabled?)
> ```

If your HTTP server is configured to use a different PHP version than the default (/usr/bin/php), `occ` should be run with the same version. For example, in CentOS 6.5 with SCL-PHP70 installed, the command looks like this:

```
sudo -u apache /opt/rh/php70/root/usr/bin/php /var/www/html/nextcloud/occ
```

> “□□
>
> Although the following examples make use of the `sudo -u ... /path/to/php /path/to/occ` method, your environment may require use of a different wrapper utility than `sudo` to execute the command as the appropriate user. Other common wrappers:
>
> - `su --command '/path/to/php ...' username` -- Note here that the target user specification comes at the end, and the command to execute is specified first.
> - `runuser --user username -- /path/to/php ...` -- This wrapper might be used in container contexts (ex: Docker / `arm32v7/nextcloud`) where both `sudo` and `su` wrapper utilities cannot be used.

Running `occ` with no options lists all commands and options, like this example on Ubuntu:

```
sudo -E -u www-data php occ
Nextcloud version 19.0.0

Usage:
 command [options] [arguments]

Options:
```

```
  -h, --help            Display this help message
  -q, --quiet           Do not output any message
  -V, --version         Display this application version
      --ansi            Force ANSI output
      --no-ansi         Disable ANSI output
  -n, --no-interaction  Do not ask any interactive question
      --no-warnings     Skip global warnings, show command output only
  -v|vv|vvv, --verbose  Increase the verbosity of messages: 1 for normal output,
                        2 for more verbose output and 3 for debug

Available commands:
  check                 check dependencies of the server
                        environment
  help                  Displays help for a command
  list                  Lists commands
  status                show some status information
  upgrade               run upgrade routines after installation of
                        a new release. The release has to be
                        installed before.
```

This is the same as `sudo -E -u www-data php occ list`.

Run it with the `-h` option for syntax help:

```
sudo -E -u www-data php occ -h
```

Display your Nextcloud version:

```
sudo -E -u www-data php occ -V
  Nextcloud version 19.0.0
```

Query your Nextcloud server status:

```
sudo -E -u www-data php occ status
  - installed: true
  - version: 19.0.0.12
  - versionstring: 19.0.0
  - edition:
```

`occ` has options, commands, and arguments. Options and arguments are optional, while commands are required. The syntax is:

```
occ [options] command [arguments]
```

Get detailed information on individual commands with the `help` command, like this example for the `maintenance:mode` command:

```
sudo -E -u www-data php occ help maintenance:mode
Usage:
 maintenance:mode [options]

Options:
      --on              enable maintenance mode
      --off             disable maintenance mode
  -h, --help            Display this help message
  -q, --quiet           Do not output any message
  -V, --version         Display this application version
      --ansi            Force ANSI output
      --no-ansi         Disable ANSI output
  -n, --no-interaction  Do not ask any interactive question
      --no-warnings     Skip global warnings, show command output only
  -v|vv|vvv, --verbose  Increase the verbosity of messages: 1 for normal output,
                        2 for more verbose output and 3 for debug
```

The `status` command from above has an option to define the output format. The default is plain text, but it can also be `json`:

```
sudo -E -u www-data php occ status --output=json
{"installed":true,"version":"19.0.0.9","versionstring":"19.0.0","edition":""}
```

or `json_pretty`:

```
sudo -E -u www-data php occ status --output=json_pretty
{
   "installed": true,
   "version": "19.0.0.12",
   "versionstring": "19.0.0",
   "edition": ""
}
```

This output option is available on all list and list-like commands: `status`, `check`, `app:list`, `config:list`, `encryption:status` and `encryption:list-modules`

# Environment variables

`sudo` does not forward environment variables by default. You can prepend the variable and use the `-E` switch to pass it through:

```
NC_debug=true sudo -E -u www-data php occ status
```

Alternatively, `export` the variable first:

```
export NC_debug=true
sudo -E -u www-data php occ status
```

# Enabling autocompletion

> “□□
>
> Command autocompletion currently only works if the user you use to execute the occ commands has a profile. `www-data` in most cases is `nologon` and therefore **cannot** use this feature.

Autocompletion is available for bash (and bash based consoles). To enable it, you have to run **one** of the following commands:

```
# BASH ~4.x, ZSH
source <(/var/www/html/nextcloud/occ _completion --generate-hook)

# BASH ~3.x, ZSH
/var/www/html/nextcloud/occ _completion --generate-hook | source /dev/stdin

# BASH (any version)
eval $(/var/www/html/nextcloud/occ _completion --generate-hook)
```

This will allow you to use autocompletion with the full path `/var/www/html/nextcloud/occ <tab>`.

If you also want to use autocompletion on occ from within the directory without using the full path, you need to specify `--program occ` after the `--generate-hook`.

If you want the completion to apply automatically for all new shell sessions, add the command to your shell's profile (eg. `~/.bash_profile` or `~/.zshrc`).

# Limitations in maintenance mode

In maintenance mode, apps are not loaded[^1], so commands from apps are unavailable. Commands integrated into Nextcloud server are available in maintenance mode.

We discourage the use of maintenance mode unless the command explicitly prompts you to do so or unless the commands' documentation explicitly states that maintenance mode should be used.

A command may use events to communicate with other apps. An app can only react to an event when loaded. Example: The command user:delete deletes a user account. UserDeletedEvent is emitted. Calendar app implements an event listener to delete user data[^2]. In maintenance mode, the Calendar app is not loaded, and hence the user data not deleted.

# Debugging

Every `occ` command accepts the standard Symfony Console verbosity flags:

- `-v` — normal output (errors, warnings, and key messages)
- `-vv` — verbose output (additional progress detail)
- `-vvv` — debug output (full trace, useful for diagnosing command failures)

Example:

```
sudo -E -u www-data php occ files:scan --all -vv
```

To also enable debug-level log output from Nextcloud itself, set the `NC_loglevel` environment variable:

```
NC_loglevel=0 sudo -E -u www-data php occ status
```

See Logging for more on log levels.

# Command reference

- Apps, background jobs & config commands

- DAV & database commands
- Encryption commands
- Files commands
- LDAP commands
- OCM commands
- User & group commands
- System & maintenance commands

[^1]: Exception: The settings app is loaded

[^2]: Calendar app event listener for UserDeletedEvent

---

□□□ #3
□ cocobook-sync □□□ 2024-09-17 22:36:43 UTC
□ cocobook-sync □□□ 2026-05-30 17:36:40 UTC