

Example installation on OpenBSD



- [HTTPD\(8\)](#)
- [PHP](#)
- [Database](#)
- [Redis](#)
- [Cron job](#)
- [Chroot](#)
- [Nextcloud final steps](#)
- [NOTE](#)



Nextcloud does not have official OpenBSD or other BSDs support

In this install tutorial we will be deploying Nextcloud on a minimal OpenBSD with our own httpd(8), PHP, PostgreSQL and redis (for -stable or -current are the same steps).

From a base installed OpenBSD system you can just do:

```
# pkg_add nextcloud
```

The extra packages:

```
# pkg_add postgresql-server redis pecl82-redis php-pdo_pgsql
```

This will take care of your dependencies and give you the options to choose which PHP version do you want.

HTTPD(8)

Create a virtualhost in `/etc/httpd.conf` and add the following content to it:

```
server "domain.tld" {
    listen on egress tls port 443
    hsts {
max-age 15768000
preload
subdomains
    }

    tls {
        certificate "/etc/ssl/domain.tld_fullchain.pem"
        key "/etc/ssl/private/domain.tld_private.pem"
    }

    # Set max upload size to 513M (in bytes)
    connection max request body 537919488
    connection max requests 1000
    connection request timeout 3600
    connection timeout 3600

    root "/nextcloud"
    directory index "index.php"

    # Ensure that no '*.php*' files can be fetched from these directories
    location "/config/*" {
        block drop
    }

    location "/data/*" {
        block drop
    }

    # Note that this matches '*.php*' anywhere in the request path.
    location "/nextcloud/*.php*" {
        fastcgi socket "/run/php-fpm.sock"
    }
}
```

```
location "/apps/*" {
    pass
}

location "/core/*" {
    pass
}

location "/.well-known/carddav" {
    block return 301 "https://$SERVER_NAME/remote.php/dav"
}

location "/.well-known/caldav" {
    block return 301 "https://$SERVER_NAME/remote.php/dav"
}

location "/.well-known/webfinger" {
    block return 301 "https://$SERVER_NAME/public.php?service=webfinger"
}

location match "/ocs-provider/*" {
    pass
}
}
```

Make sure that httpd(8) is enabled and started:

```
# rcctl enable httpd
# rcctl start httpd
```

PHP

Assuming that you are on OpenBSD -current (or >= 6.8-stable) you could use PHP 8.2 so I will keep this version, but the concept is the same for other version.

The PHP packages will be available since you installed Nextcloud with pkg_add, so you just need to adjust a bit your php.ini.

It is recommended to add opcache to it:

```
[opcache]
opcache.enable=1
opcache.memory_consumption=512
opcache.interned_strings_buffer=8
opcache.max_accelerated_files=10000
opcache.revalidate_freq=1
opcache.save_comments=1
```

And increase some limits:

```
post_max_size = 513M
upload_max_filesize = 513M
```

We can enable the PHP modules with:

```
# cd /etc/php-8.2.sample
# for i in *; do ln -sf ../php-8.2.sample/$i ../php-8.2/; done
```

And then we just enable and start PHP:

```
# rcctl enable php82_fpm
# rcctl start php82_fpm
```

Database

As mentioned, we will be using PostgreSQL as our database, and we already installed it, now we need to initialise:

```
$ su - _postgresql
$ mkdir /var/postgresql/data
$ initdb -D /var/postgresql/data -U postgres -A md5 -E UTF8 -W
...
Enter new superuser password: PASSWORD
Enter it again: PASSWORD
...
Success. You can now start the database server using:
```

```
pg_ctl -D /var/postgresql/data -l logfile start

$ pg_ctl -D /var/postgresql/data -l logfile start
server starting
$ exit
```

We need to check, enable and start postgres:

```
# rcctl check postgresql
# rcctl enable postgresql
# rcctl start postgresql
```

You can follow the README on `/usr/local/share/doc/pkg-readmes/postgresql-server` to create users and permission.

Redis

We installed redis before, we need to enable it and start it and also add it to the Nextcloud conf:

```
# rcctl enable redis
# rcctl start redis
# mg /var/www/nextcloud/config/config.php
...
'memcache.local' => '\OC\Memcache\Redis',
'redis' => array(
'host' => 'localhost',
'port' => 6379,
'timeout' => 0.0,
),
...
```

Cron job

We need to add the Nextcloud cron job to get some tasks done by adding this entry on your cronjob:

```
*/* * * * * /usr/bin/ftp -Vo - https://domain.tld/cron.php >/dev/null
```

Chroot

Since in OpenBSD `httpd(8)` works with a `chroot(8)` by default, we need to be sure that we have the relevant files into the `/var/www` jail:

```
# mkdir -p /var/www/etc/ssl
# install -m 444 -o root -g bin /etc/ssl/cert.pem /etc/ssl/openssl.cnf \
    /var/www/etc/ssl/
# cp /etc/resolv.conf /var/www/etc
```

Nextcloud final steps

The remaining installation steps are completed in the web-based installation wizard.

To activate this wizard, create a file named `CAN_INSTALL` inside the installation's config folder:

```
## # touch /var/www/nextcloud/config/CAN_INSTALL
```

Use your browser to navigate to the installation's URL:

```
## https://domain.tld
```

Now you just need to follow the steps and put in place your DB name, user and passwords.

Keep in mind that the upgrades for Nextcloud you can do it by running on `-current`:

```
# pkg_add -u -Dsnap
```

And on `-stable`:

```
# pkg_add -u
```

Then you just follow the steps from your browser.

NOTE

Remember always to read all the READMEs from the OpenBSD packages on:

```
/usr/local/share/doc/pkg-readmes/
```

All this information and more is available for you there.

#1

cocobook-sync 2021-02-18 16:20:43 UTC

cocobook-sync 2026-05-06 16:21:16 UTC