Serveur Nextcloud

[](https://doc.ubuntu-fr.org/_media/utilisateurs/filerem1/nextcloud/nextcloud_logo.small.png)

[Nextcloud](https://doc.ubuntu-fr.org/nextcloud) est une solution de stockage et de [partage](https://doc.ubuntu-fr.org/partage) de fichiers en ligne.

Nextcloud permet également de synchroniser agendas et carnets d'adresses. Il existe de nombreuses autres fonctionnalités disponibles grâce à des modules externes.

Sa grande force réside dans le fait que vous contrôlez entièrement vos données, l'emplacement du stockage, le partage, la suppression, etc. Tout reste sous votre contrôle, à la différence des services comme *Dropbox* ou *Google Drive*, dont on ne peut être sûr de ce qu'ils font de vos données.

Cette page traite de l'installation et de la configuration d'un **serveur Nextcloud**. Pour une approche plus générale de ce projet consultez [cette page de la documentation](https://doc.ubuntu-fr.org/nextcloud).

Nextcloud étant une solution de cloud centralisé, elle requiert la mise en place d'un serveur. Si vous ne vous sentez pas l'âme d'un administrateur système vous pouvez profiter des services d'[un prestataire](https://doc.ubuntu-fr.org/nextcloud#prestataires) qui s'occupera de cet aspect particulièrement technique.

Nous allons voir ici comment déployer et utiliser ce service sur un serveur privé (ou éventuellement mutualisé).

Haut du formulaire

Modifier

Bas du formulaire

Pré-requis

* Disposer d'un [serveur](https://doc.ubuntu-fr.org/serveur)
* Avoir un [serveur Web avec PHP](https://doc.ubuntu-fr.org/lamp) installé dessus (ici [Apache2](https://doc.ubuntu-fr.org/apache2)), avoir activé le [mode SSL d'apache](https://doc.ubuntu-fr.org/tutoriel/securiser_apache2_avec_ssl#mise_en_application_avec_le_serveur_http_apache2) (https), avoir un [serveur ssh](https://doc.ubuntu-fr.org/openssh) (conseillé), ainsi qu'une base de données (ici [mysql](https://doc.ubuntu-fr.org/mysql" \o "mysql)).
* Savoir ajouter une tâche [CRON](https://doc.ubuntu-fr.org/cron)
* Savoir installer et activer des [modules php](https://doc.ubuntu-fr.org/php#modules) et [apache](https://doc.ubuntu-fr.org/apache2).
* Savoir créer une base de données [MySQL](https://doc.ubuntu-fr.org/mysql).
* Avoir un PC client avec un navigateur web et un client SSH.
* Disposer des [droits d'administration](https://doc.ubuntu-fr.org/sudo) sur le serveur.
* Disposer d'une connexion à Internet configurée et activée.
* Savoir utiliser le [terminal](https://doc.ubuntu-fr.org/terminal)

Bas du formulaire

Installation

Haut du formulaire

Bas du formulaire

Téléchargement et Extraction de Nextcloud

Le téléchargement se fait via le site officiel : <https://nextcloud.com/install/#instructions-server>

L’installateur se présente sous la forme d'une archive [ZIP](https://doc.ubuntu-fr.org/zip), à dézipper à la racine de votre serveur web. L'extraction va créer un dossier nextcloud. Déplacez son contenu à la racine du serveur, et supprimez le dossier vide. Rien d'autre ne doit être présent à la racine de votre serveur. Nextcloud vérifie l’intégrité des fichiers, et l'analyse échouera s'il reste d'autres éléments.

Haut du formulaire

Modifier

Bas du formulaire

Accès à l'installateur

[](https://doc.ubuntu-fr.org/_media/utilisateurs/filerem1/nextcloud/nextcloud_module_php_install.png)

Rendez-vous avec votre navigateur sur la page web de votre serveur, soit par son domaine si vous l'avez configuré, soit par son adresse IP.

Si vous avez réalisé les opérations précédentes en tant qu'administrateur (donc *owner* et *group* à *root*), exécutez:

$ sudo chown www-data:www-data -R nextcloud/

(remplacez nextcloud/ par votre dossier racine)

Haut du formulaire

Modifier

Bas du formulaire

Installation des modules PHP manquant

La page d'installation indique qu'il manque des modules PHP. Connectez vous en [SSH](https://doc.ubuntu-fr.org/ssh) avec les [droits d'administration](https://doc.ubuntu-fr.org/sudo) sur votre serveur, puis [installez ces paquets](https://doc.ubuntu-fr.org/tutoriel/comment_installer_un_paquet#en_ligne_de_commande) :

$ apt install php-zip php-gd php-curl php-intl

Les installer ne suffit pas, il faut les activer, et redémarrer le serveur web pour que ceux ci soient pris en compte :

$ phpenmod zip gd curl intl

puis on redémarre le serveur Apache :

$ service apache2 restart

Revenez sur le navigateur web et rafraîchissez la page. Vous devriez maintenant voir l'étape suivante.

Haut du formulaire

Modifier

Bas du formulaire

Identifiant, emplacement de stockage et base de données

[](https://doc.ubuntu-fr.org/_media/utilisateurs/filerem1/nextcloud/nextcloud_install_server-2.png)

À cette étape, vous allez créer le fichier de configuration du serveur nextcloud.

* Choisissez votre identifiant et mot de passe d'administration. C'est ce compte qui servira à créer d'autres comptes, à mettre à jour, à installer des applications, etc.
* Choisissez le répertoire des données. Vous pouvez laissez tel quel ou bien changer l'emplacement, par exemple si vous avez un disque supplémentaire, ou distant. Si vous changez l'emplacement, il vous faudra [donner les droits](https://doc.ubuntu-fr.org/droits) au serveur web pour pouvoir y accéder. Sinon, vous pourrez accéder au serveur mais vous ne pourrez rien y stocker. Veillez aussi à ce que ces emplacements soient [monté au démarrage](https://doc.ubuntu-fr.org/montage) avec les bons droits.
* Choisissez la base de données. Avant de remplir ce champ, il faut que la base de données existe déjà. Dans le cas de Mysql, nous allons la créer en ligne de commande via SSH. Dans votre session SSH en administrateur, exécutez ces commandes :

$ mysql

$ >CREATE DATABASE nom\_de\_la\_base; # Remplacez nom\_de\_la\_base par ce que vous voulez. N'oubliez pas le ; puis validez.

$ >show databases; # Ceci vérifie que la base à bien été créée. Si vous la voyez dans la liste, parfait! Sinon, renseignez vous sur les problèmes de création de base mysql.

* **Remplissez maintenant l'encart base de données :**

1. Utilisateur de la base de donnée ;
2. Mot de passe de la base de donnée ;
3. Le nom de la base de donnée ;
4. L'emplacement de la base donnée et le numéro du port, par défaut localhost (signifie qu'elle est sur la même machine que le serveur web)

* Cliquer ensuite sur **"Terminer l'installation"** , l'installation prend un moment en fonction de la puissance de votre serveur. Si vous tombez sur une **"Erreur 500"**, rechargez la page jusqu'à ce que l'installation finisse (F5).
* Lorsque la page de connexion apparaît, l'installation est terminée. Procédez ensuite au réglage des performances du serveur Nextcloud.

Haut du formulaire

Bas du formulaire

Applications

Une fois installé, vous disposez des partages de base: *Fichiers*, *Galerie*  
Il est possible d'en ajouter d'autre, notamment: ***Agenda*, *Contacts*, *Task (tâches)***, … partage social : *Diaspora*, Chat voir visioconférence avec *Talk*…  
Pour cela, cliquez sur votre icône en haut à droite puis dans le menu, ***+ Applications***  
Dans la liste de gauche, vous avez ***Pack d'applications***, dans la liste, il suffit de cliquer le bouton sur la droite ***Activer***

Haut du formulaire

Bas du formulaire

Migration d'un serveur OwnCloud en serveur Nextcloud

Il est facile de migrer un serveur OwnCloud en serveur Nextcloud. La page de référence est : [doc Nextcloud pour la migration (en)](https://help.nextcloud.com/t/migrating-from-owncloud-to-nextcloud/551)  
Il faut commencer par récupérer la version Nextcloud compatible avec votre version d'OwnCloud:

* ownCloud 8.2 et 9.0 vers Nextcloud 9.
* ownCloud 9.0 et 9.1 vers Nextcloud 10
* ownCloud 10.0.\* vers Nextcloud 12.0.\*

Puis suivre les étapes :

Comme toujours, pensez à sauvegarder au préalable, notamment la base de données qui va être modifiée et donc devenir incompatible avec OwnCloud ! Pour les données, il est préférable d'avoir un répertoire indépendant, si ce n'est déjà fait, c'est l'occasion : le fichier *config.php* contient le paramètre:

'datadirectory' => '/path/my\_owncloud\_data',

Évidemment, il est préférable de réaliser cette opération à des heures où les utilisateurs ne sont pas connectés (et de les prévenir au préalable)!

1. À part les répertoires *config* et *data*, supprimer le reste du contenu de votre répertoire OwnCloud.
2. Décompresser les fichiers et répertoire de Nextcloud dans le répertoire où étaient les fichiers OwnCloud.
3. Lancer la migration en appelant la page de votre cloud : [*https://mon\_cloud.exemple.fr/*](https://mon_cloud.exemple.fr/)*ou mon\_serveur/mon\_cloud*et laissez-vous guider. Pour avoir un serveur Nextcloud à jour, vous pouvez enchaîner les mises à jour Nextcloud à dans l'administration de Nextcloud

Haut du formulaire

Bas du formulaire

Réglage des performances

Connectez-vous avec vos identifiant et mot de passe choisis à l'installation, puis cliquez sur la roue dentée https://doc.ubuntu-fr.org/_media/utilisateurs/filerem1/nextcloud/nextcloud_icone_parametre-2.png?w=20&tok=75e60b en haut à droite, sur le bandeau bleu, et enfin sur "**Administration/Paramètres de Base**"

Vous voyez en haut de la page le titre **"Avertissement de sécurité & configuration"**, nous allons nous occuper de tout cela.

Haut du formulaire

Bas du formulaire

Redirection HTTPS automatique

Par défaut, même après avoir activé le mode SSL d'apache et déclaré votre site, l'accès en HTTP fonctionne toujours. Nous allons paramétrer une redirection automatique de HTTP vers HTTPS, comme le[préconise la documentation Nextcloud](https://docs.nextcloud.com/server/9/admin_manual/configuration_server/harden_server.html#use-https).

Partie à revoir

Ouvrez votre session SSH en administrateur, et modifiez le fichier de configuration de votre site qui se trouve dans ce dossier :

$ nano /etc/apache2/sites-available/"votre-site.conf" # remplacer "votre-site.conf par le nom correct de votre fichier de configuration

Rajouter en dessous de la ligne **"ServerName cloud.nextcloud.com"**:

ServerSignature Off

RewriteEngine On

RewriteRule ^ https://%{SERVER\_NAME}%{REQUEST\_URI} [END,QSA,R=permanent]

Un petit **CTRL + O** pour enregistrer, un **CTRL + X** pour quitter. On active le module **rewrite** d'apache2 pour que ces lignes soient prise en compte :

$ a2enmod rewrite

Puis on redémarre le serveur web pour prendre en compte la configuration:

$ service apache2 restart

On rafraîchit la page web. Si vous n'avez pas de certificat validé par une autorité, cela vous occasionnera une erreur SSL. Passez outre la recommandation, vous pourrez changer de certificat SSL plus tard.

Haut du formulaire

Bas du formulaire

Utilisation HTTP/2 pour le HTTPS

Si votre site est configuré pour le HTTPS, il est conseillé d'activer également le protocole HTTP/2. Il faut commencer par activer le module http2 d'Apache.

Ce module n'est pas disponible par défaut dans Ubuntu 16.04. Vous pouvez cependant l'installer, à vos risques et périls, via la procédure décrite ici : [Comment configurer Apache avec le support HTTP / 2 sur Ubuntu 16.04](https://websetnet.net/fr/set-apache-http2-support-ubuntu-16-04/)

$ a2enmod http2

Rajoutez les lignes suivantes à votre fichier de configuration d'hôte virtuel HTTPS :

<IfModule mod\_http2.c>

Protocols h2 http/1.1

</IfModule>

Puis on redémarre le serveur web pour prendre en compte la configuration:

$ systemctl restart apache2

Haut du formulaire

Modifier

Bas du formulaire

Activer HSTS ( HTTP Strict-Transport-Security )

Toujours en accord avec les [recommandations officielles](https://docs.nextcloud.com/server/9/admin_manual/configuration_server/harden_server.html#enable-http-strict-transport-security), "HTTP Strict Transport Security" ([wikipedia](https://fr.wikipedia.org/wiki/HTTP_Strict_Transport_Security" \o "https://fr.wikipedia.org/wiki/HTTP_Strict_Transport_Security)) est un mécanisme de sécurité supplémentaire contre les attaques de type man-in-the-middle. Il est recommandé de l'activer.

Pour ce faire, nous allons retourner dans notre session SSH et modifier le fichier de configuration d'hôte virtuel HTTPS.

Rajouter en dessous de "**ServerName**" :

<IfModule mod\_headers.c>

Header always set Strict-Transport-Security "max-age=15768000; includeSubDomains; preload"

</IfModule>

On enregistre, on quitte l'éditeur, puis on active le module correspondant dans apache, pour que ces lignes soient prises en compte :

a2enmod headers

On redémarre le serveur apache pour que la configuration soit prise en compte, puis on recharge la page web du serveur Nextcloud. La ligne traitant de "HTTP Strict-Transport-Security" à disparu, nous pouvons continuer.

Haut du formulaire

Modifier

Bas du formulaire

Cache mémoire

Le cache mémoire est conseillé pour améliorer la réactivité de votre serveur. Les fichiers les plus demandés sont mis en cache mémoire pour une plus grande disponibilité.

Dans la [documentation officielle](https://docs.nextcloud.com/server/9/admin_manual/configuration_server/caching_configuration.html#configuring-memory-caching), le cache php-apc est désigné comme obsolète. Elle préconise d'utiliser [REDIS](https://docs.nextcloud.com/server/9/admin_manual/configuration_server/caching_configuration.html#id4) ou [APCu](https://docs.nextcloud.com/server/9/admin_manual/configuration_server/caching_configuration.html" \l "id2" \o "https://docs.nextcloud.com/server/9/admin_manual/configuration_server/caching_configuration.html#id2) que nous utiliserons ici.

Reprenez votre session SSH, allez à la racine de votre serveur nextcloud **( en général dans /var/www/html )**

On installe et on active le module php-apcu

$ apt install php-apcu

$ phpenmod apcu

Puis on modifie le fichier config.php se trouvant dans le dossier config du serveur.

Rajoutez une ligne à la fin, avant la dernière parenthèse :

'memcache.local' => '\OC\Memcache\APCu',

Enregistrez, redémarrez le serveur web, puis rechargez la page web Nextcloud : la ligne concernant le cache est partie.

Haut du formulaire

Bas du formulaire

PHP OPcache

L'OPcache est une recommandation de la [documentation](https://docs.nextcloud.com/server/12/admin_manual/configuration_server/server_tuning.html#enable-php-opcache). Il est [complémentaire](https://mnt-tech.fr/blog/opcache-configuration-et-fonctionnement/) du cache mémoire Apcu ou Redis.  
Nous allons éditer le fichier php.ini, pour activer la prise en charge d'OPcache.

$ nano /etc/php/7.x/apache2/php.ini # remplacez 7.x par votre version de php

rajouter, tout en bas du fichier :

opcache.enable=1

opcache.enable\_cli=1

opcache.interned\_strings\_buffer=8

opcache.max\_accelerated\_files=10000

opcache.memory\_consumption=128

opcache.save\_comments=1

opcache.revalidate\_freq=1

On enregistre, on quitte, on redémarre le serveur web, et on recharge la page web Nextcloud : tous les tests on réussi, le serveur est correctement configuré. Pour autant, il reste une dernière configuration à effectuer : la tâche [CRON](https://doc.ubuntu-fr.org/cron).

Haut du formulaire

Modifier

Bas du formulaire

Configurer le CRON

Nextcloud utilise une tâche *cron* pour lancer des scripts à intervalles réguliers. Par défaut, il est configuré pour utiliser [AJAX](https://fr.wikipedia.org/wiki/Ajax_(informatique)), mais qui à le désavantage de ne fonctionner que via l'explorateur web. Pour une utilisation via l'application, il est [recommandé](https://docs.nextcloud.com/server/12/admin_manual/configuration_server/background_jobs_configuration.html#cron) d'utiliser le [CRON](https://doc.ubuntu-fr.org/cron) de votre serveur, paramétré avec l'utilisateur qui gère votre site (par défaut www-data)

Nous allons appeler le fichier cron.php se trouvant à la racine de votre serveur web toutes les 5 minutes, par l'utilisateur www-data. ( à adapter selon votre configuration serveur)

$ crontab -u www-data -e

À la dernière ligne, rajouter le cron de 5 minutes : /!\Attention de bien modifier le chemin vers votre fichier cron.php ex: /var/www/html/nextcloud/cron.php, ou tout autre chemin selon votre configuration, sinon le cron ne fonctionera pas. /!\

\*/5 \* \* \* \* php -f /var/www/cron.php

Quittez en enregistrant, puis vérifiez la configuration par

$ crontab -u www-data -l

\*/5 \* \* \* \* php -f /var/www/cron.php

Votre serveur Nextcloud est maintenant installé et configuré selon les recommandations de l'éditeur. Pour configurer le client, voir la page dédié au [Client Nextcloud](https://doc.ubuntu-fr.org/nextcloud-client)

Pour vérifier le bon fonctionnement de cron depuis votre interface d'administration Nextcloud, vous pouvez forcer le redémarrage de cron afin que le fichier cron.php soit appeler directement et ne pas devoir attendre le délais de 15min, pour cela redémarrer cron avec

$ sudo service cron restart

Si tout est correct, vous obtiendrez le message suivant depuis l'interface d'administration Nextcloud :

"Tâches de fond | Dernière tâche exécutée il y a 1 minute" (ou inférieur)

Haut du formulaire

Bas du formulaire

Paramétrage Divers

Haut du formulaire

Bas du formulaire

Langue

Vous pouvez définir une langue par défaut à chaque création d'utilisateur, et les paramêtre régionnaux correspondant.

Pour ce faire, rajoutez ces lignes dans le fichier config/config.php :

'default\_language' => 'fr\_FR',

'force\_language' => 'fr\_FR',

'default\_locale' => 'fr\_FR',

'force\_locale' => 'fr\_FR',

Adaptez cette valeur à la langue que vous souhaitez.

Source : [Nextcloud Language Configuration](https://docs.nextcloud.com/server/12.0/admin_manual/configuration_server/language_configuration.html" \o "https://docs.nextcloud.com/server/12.0/admin_manual/configuration_server/language_configuration.html)

Haut du formulaire

Bas du formulaire

Thème

Pour que l'interface web adapte les icônes automatiquement à vos couleur, installez ceci :

$ sudo apt install php-imagick libmagickcore-6.q16-3-extra

On Active le module php :

$ phpenmod imagick

puis on redémarre le serveur Apache :

$ service apache2 restart