



PROJET ENCADRÉ PAR

Mr Gnadjro

MISE EN PLACE



NEXTCLOUD



PRÉSENTÉ PAR

BÉLINDA KOPOIN
JEAN RICHARDSON JR
SOFIANE KANOUNI
SAFA CHAABEN

“TAKE CONTROL OF YOUR DATA”

Sommaire:

I- INTRODUCTION.....	3
II- FONCTIONNALITÉS PRINCIPALES.....	3
III- SPÉCIFICATIONS TECHNIQUES	5
IV- CAS D'UTILISATION.....	6
V- AVANTAGES CLÉS.....	6
VI- INSTALLATION ET CONFIGURATION.....	8
VII- EXEMPLES DE DÉPLOIEMENT.....	10

I- INTRODUCTION

Contexte:

Suite à une analyse du système d'information, plusieurs problèmes liés à l'utilisation des outils informatiques ont été identifiés. Conscient de ces dysfonctionnements, M. Étienne MALLET, président actuel de la société, nous a contactés pour nous confier une mission clé : rénover le réseau informatique afin que l'architecture soit adaptée aux exigences de l'entreprise.

Pour cela, notre mission consistera à :

1. Réaliser un audit approfondi du SI de l'entreprise ;
2. Mutualiser certaines ressources en fonction des services (fichiers utilisateurs, imprimantes);
3. Optimiser la gestion des pannes afin de les rendre plus efficaces. Ceci permettra aux utilisateurs d'envoyer des demandes d'intervention directement depuis leur ordinateur, qu'il soit personnel ou professionnel ;
4. Mettre en place un inventaire informatisé du parc informatique afin de contrôler les licences logicielles, les mises à jour système, les systèmes d'exploitation installés, etc. ;
5. Proposer un système de sauvegarde personnalisé de type « cloud privé » pour chaque utilisateur;
6. Assurer la continuité de service du système informatique.

Dans cette partie, nous nous concentrerons sur la mise en place d'une solution de type cloud privé à l'aide du logiciel Nextcloud.

Définition:

NEXTCLOUD est un logiciel *cloud gratuit et en open source* créé en juin 2016 par Frank Karlitschek. Il permet l'auto-hébergement et le Cloud computing; cette solution est utilisée aussi bien pour les entreprises que pour les particuliers.

De plus, il permet de collaborer avec toute personne ayant accès au cloud et d'y stocker tous les fichiers en toute sécurité.

II- FONCTIONNALITÉS PRINCIPALES

Nextcloud est un logiciel flexible grâce aux nombreuses fonctionnalités et extensions qu'il propose. Les fonctionnalités les plus utilisés sont :

- Active Directory :
- Suivi des activités:
- Tableau de bord Nextcloud: Une sorte de planning:
- Prise en charge du RGPD, CCPA , HIPAAA, FERPA, COPPA, MTCDDPA:
- Cryptage de bout en bout :
- Cryptage côté serveur avec "Gestion souple des clés"
- Nextcloud Federation (Partage de fichier, chats, appels)
- Prévention de la perte de données
- 2FA
- Protection contre la force brute

III- SPÉCIFICATIONS TECHNIQUES

Afin de mettre en place notre cloud privé, il sera nécessaire d'avoir à sa disposition, au minimum, un système **LAMP** (Linux, Apache, MySQL/MariaDB, PHP). Vous trouverez les spécifications détaillées ci-joint:

1. Serveur : Système d'exploitation : Linux (Ubuntu, Debian, CentOS, etc.).
2. Base de données : MySQL/MariaDB, PostgreSQL ou SQLite.
3. Serveur web : Apache ou Nginx.
4. PHP : Version 7.4 ou supérieure.

De Plus, nous utiliserons des protocoles tels que:

- WebDAV (extension HTTP) pour la synchronisation.
- OpenID Connect et SAML pour l'authentification.

Pour le bon fonctionnement du cloud privé, les salariés devront disposer d'un navigateur compatible (Chrome, Firefox, Edge) ainsi que d'une application (Windows, macOS, Linux, Android, iOS).

Ainsi, dans notre cas, nous utiliserons :

- Serveur web : Debian.
- Base de données : MariaDB.
- Serveur web : Apache.
- Version PHP : 8.2.

Ce serveur web sera accessible pour les salariés par l'application Linux et sera compatible avec tous les navigateurs. Il utilisera les protocoles WebDAV et OpenID Connect / SAML.

IV- CAS D'UTILISATION

Voici différents cas d'utilisation du logiciel Nextcloud

Cas d'entreprise :

- *Stockage centralisé* permettant à toutes les équipes de s'organiser efficacement et rapidement.
- *Collaboration sécurisée* avec des partenaires externes via les protocoles TLS/SSL qui chiffrent les données lors des communications

Particuliers :

- *Stockage de données personnel*, sécurisé et auto-hébergé.

V- AVANTAGES CLÉS

Nextcloud est une solution complète offrant une gamme de fonctionnalités adaptées aux besoins des particuliers et des entreprises, parmi lesquelles :

Synchronisation multi-appareils : Accédez à vos fichiers cloud depuis un ordinateur ou un appareil mobile grâce à une synchronisation fluide entre tous vos dispositifs.

Partage sécurisé : Gérez les permissions d'accès pour chaque utilisateur, garantissant ainsi la confidentialité et le contrôle des données stockées.

Collaboration en temps réel : Travaillez simultanément sur un même document avec plusieurs collaborateurs pour une productivité accrue.

Messagerie instantanée et visio conférence : Avec Nextcloud Talk, profitez d'une plateforme de communication professionnelle intégrant messagerie, appels vocaux, vidéoconférences et partage de fichiers, directement au sein de Nextcloud Hub.

Chiffrement des données : Grâce au cryptage côté client, Nextcloud assure une protection renforcée contre les attaques ciblant le serveur, garantissant une sécurité optimale des données.

Authentification à double facteur (2FA) : Ajoutez une couche de sécurité supplémentaire pour protéger les comptes utilisateurs contre toute tentative de piratage.

Principaux avantages de Nextcloud :

1. Contrôle total des données :

- En hébergeant vos fichiers sur vos propres serveurs, vous conservez un contrôle absolu sur leur gestion, sans dépendance à des tiers.

2. Alternative open source aux solutions propriétaires :

- Contrairement à Google Drive ou Dropbox, Nextcloud est open source et offre une version communautaire gratuite. Cette approche garantit une transparence, une sécurité accrue et une liberté totale pour les utilisateurs.

3. Coûts réduits :

- Nextcloud constitue une solution économique par rapport aux services SaaS (Software as a Service) traditionnels.

4. Conformité RGPD et HIPAA :

- Le RGPD (Règlement Général sur la Protection des Données) impose des normes strictes pour protéger la confidentialité des données dans l'Union européenne, en vigueur depuis le 25 mai 2018.

La loi américaine HIPAA établit des exigences spécifiques pour sécuriser et protéger les données de santé, répondant aux besoins des organisations du secteur médical.

Tarification et versions :

Version communautaire : Open source et gratuite, idéale pour les utilisateurs souhaitant une solution autonome.

Offres commerciales : Nextcloud propose des solutions payantes pour les entreprises, incluant un support professionnel et des fonctionnalités avancées via Nextcloud Enterprise.

VI- INSTALLATION ET CONFIGURATION

Etape 1: Installation et configuration du serveur web

Installation et configuration du serveur Apache

tout d'abord on se connecte en mode root pour effectuer toutes les commandes à venir, on se connecte en mode superutilisateur

```
safa@debian:~$ su -  
Mot de passe :  
root@debian:~# S
```

Ensuite on vérifie et on lance les mises à jour de paquet

```
root@debian:~# apt-get update  
Atteint :1 http://security.debian.org/debian-security bookworm-security InRelease  
Atteint :2 http://deb.debian.org/debian bookworm InRelease  
Atteint :3 http://deb.debian.org/debian bookworm-updates InRelease  
Lecture des listes de paquets... Fait  
root@debian:~# apt-get upgrade  
Lecture des listes de paquets... Fait  
Construction de l'arbre des dépendances... Fait  
Lecture des informations d'état... Fait  
Calcul de la mise à jour... Fait  
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.  
root@debian:~#
```

Ensuite on installe Apache, PHP 8.2 et leurs modules nécessaires, puis active des configurations et modules spécifiques pour assurer le fonctionnement de Nextcloud :

```

root@debian:~# apt-get -y install apache2 curl apache2-doc apache2-utils libapache2-mod-php8.2 php8.2 php
8.2-common php8.2-gd php8.2-mysql php8.2-imap php8.2-cli libapache2-mod-fcgid apache2-suexec-pristine php
-pear mcrypt imagemagick libruby libapache2-mod-python php8.2-curl php8.2-intl php8.2-pspell php8.2-sqlit
e3 php8.2-tidy php8.2-xmllrpc php8.2-xsl php-imagick php8.2-zip php8.2-mbstring php8.2-soap php8.2-fpm php
8.2-opcache php8.2-apcu
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait

```

On active les modules Apache nécessaires pour gérer les fonctionnalités avancées de Nextcloud, telles que la réécriture d'URL, la prise en charge SSL, les en-têtes HTTP, les actions FastCGI, et la gestion des fichiers WebDAV :

```

root@debian:~# a2enmod suexec rewrite ssl include dav_fs auth_digest expires
headers fastcgi actions proxy_fcgi
Module suexec already enabled
Module rewrite already enabled
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Module socache_shmcb already enabled
Module ssl already enabled
Considering dependency mime for include:
Module mime already enabled

```

```

root@debian:~# systemctl restart apache2
root@debian:~#

```

Cette commande configure Apache pour utiliser PHP 8.2 avec FPM (FastCGI Process Manager), optimisant l'exécution des scripts PHP :

```

root@debian:~# a2enconf php8.2-fpm
Enabling conf php8.2-fpm.
To activate the new configuration, you need to run:
    systemctl reload apache2
root@debian:~# systemctl reload apache2
root@debian:~#

```

Nous allons créer un groupe système et un utilisateur système, ce qui permettra de les associer à notre Virtual Host via PHP-FPM. Cela offrira une meilleure isolation des processus et une gestion optimisée des droits. Étant donné que Nextcloud est un serveur de stockage avec des besoins importants en écriture sur les fichiers et dossiers, l'utilisation de PHP-FPM s'avère plus efficace pour répondre à ces exigences

Tout d'abord on va donc créer un groupe et un utilisateur :

```
root@debian:~# groupadd web001
root@debian:~# useradd -M -g web001 -s /usr/sbin/nologin web001
root@debian:~#
```

On va pouvoir maintenant créer la structure de notre dossier de virtualHost (une configuration utilisée par un serveur web (ici Apache) pour héberger plusieurs sites ou services sur une même machine)

Nous allons placer notre projet ici :

```
root@debian:~# cd /var/www/
root@debian:/var/www# ls
html
root@debian:/var/www#
```

On crée une structure de répertoires pour Nextcloud : un dossier pour la racine du projet (**web**), un pour les fichiers temporaires (**tmp**), et un pour les journaux (**log**). Ensuite, on attribue les droits d'accès à un utilisateur et un groupe système dédiés (par exemple, **web001**). Cela garantit une meilleure isolation, une gestion sécurisée des droits, et une préparation optimale pour PHP-FPM.

```
root@debian:/var/www# mkdir -p /var/www/nextcloud/web
root@debian:/var/www# chown -R web001:web001 /var/www/nextcloud/web
```

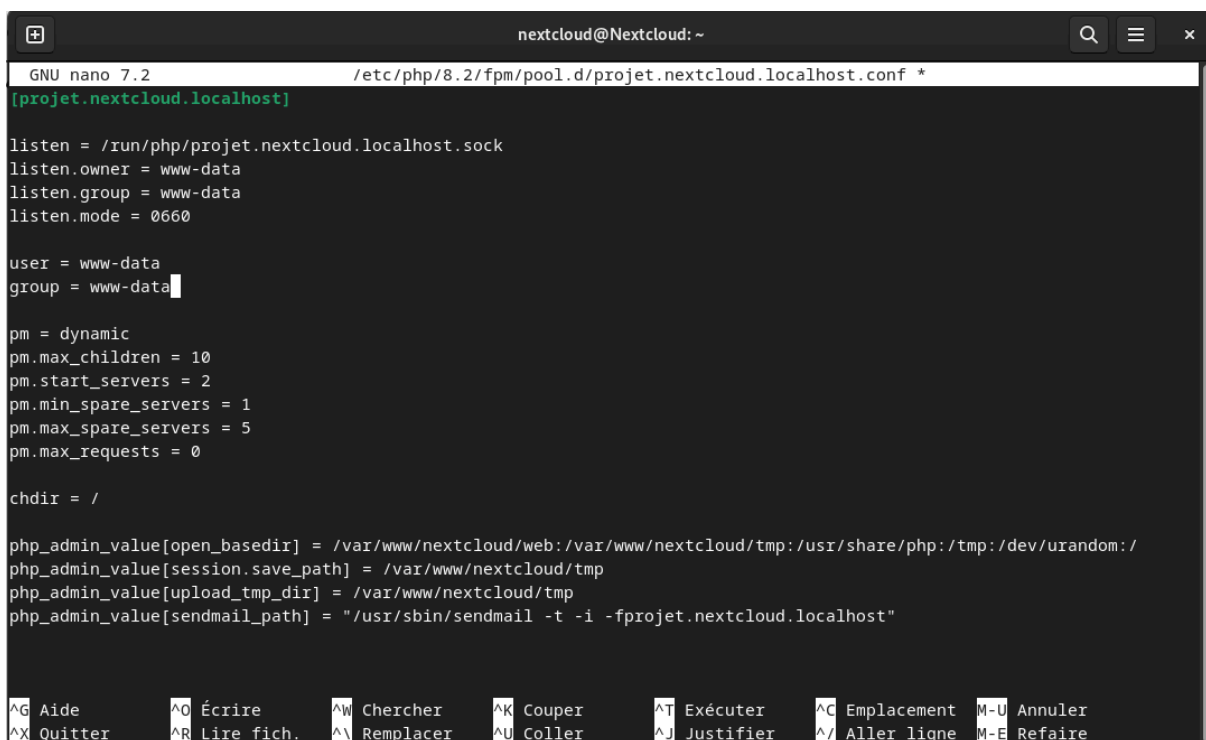
```
root@debian:/var/www# mkdir -p /var/www/nextcloud/tmp
root@debian:/var/www# chown -R web001:web001 /var/www/nextcloud/tmp
root@debian:/var/www# mkdir -p /var/www/nextcloud/log
```

Notre structure est prête ! Nous allons maintenant pouvoir configurer nos fichiers PHP et Apache, afin de faire fonctionner notre VirtualHost :

On va créer un POOL pour php FPM qui va nous permettre de faire une liaison entre Apache et PHP FPM

```
root@debian:/var/www# nano /etc/php/8.2/fpm/pool.d/projet.nextcloud.localho
st.conf
```

Ce fichier configure PHP-FPM pour Nextcloud en spécifiant comment PHP doit gérer les processus, les fichiers temporaires, les permissions d'accès. Il optimise le nombre de processus PHP, le répertoire où les sessions et fichiers temporaires sont stockés, et en restreignant les accès pour des raisons de sécurité.



```
nextcloud@Nextcloud: ~
GNU nano 7.2 /etc/php/8.2/fpm/pool.d/projet.nextcloud.localhost.conf *
[projet.nextcloud.localhost]

listen = /run/php/projet.nextcloud.localhost.sock
listen.owner = www-data
listen.group = www-data
listen.mode = 0660

user = www-data
group = www-data

pm = dynamic
pm.max_children = 10
pm.start_servers = 2
pm.min_spare_servers = 1
pm.max_spare_servers = 5
pm.max_requests = 0

chdir = /

php_admin_value[open_basedir] = /var/www/nextcloud/web:/var/www/nextcloud/tmp:/usr/share/php:/tmp:/dev/urandom:/
php_admin_value[session.save_path] = /var/www/nextcloud/tmp
php_admin_value[upload_tmp_dir] = /var/www/nextcloud/tmp
php_admin_value[sendmail_path] = "/usr/sbin/sendmail -t -i -fprojet.nextcloud.localhost"

^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C Emplacement M-U Annuler
^X Quitter   ^R Lire fich.^_ Remplacer  ^U Coller    ^J Justifier ^_ Aller ligne M-E Refaire
```

On passe maintenant au fichier de configuration Apache et la création du Virtual Host :

```
root@debian:/var/www# nano /etc/apache2/sites-available/projet.nextcloud.localhost.conf
```

Contenu du Fichier

```
GNU nano 7.2 /etc/apache2/sites-available/projet.nextcloud.localhost.conf
<Directory /var/www/nextcloud>
    AllowOverride None
    Require all denied
</Directory>

<VirtualHost *:80>
    DocumentRoot /var/www/nextcloud/web

    ServerName projet.nextcloud.localhost
    ServerAdmin webmaster@projet.nextcloud.localhost

    CustomLog /var/www/nextcloud/log/access.log combined
    ErrorLog /var/www/nextcloud/log/error.log

    SetEnvIf Authorization "(.*)" HTTP_AUTHORIZATION=$1

    <Directory /var/www/nextcloud/web>
        Options +FollowSymLinks -Indexes
        AllowOverride All
        Require all granted
    </Directory>

    <Directory /var/www/nextcloud/web>
        <FilesMatch ".+\.php(p[3457]?|t|tml)$">
            SetHandler "proxy:unix:/run/php/projet.nextcloud.localhost.sock|>
        </FilesMatch>
    </Directory>
```

```
    <Directory /var/www/nextcloud/web>
        <FilesMatch ".+\.php(p[3457]?|t|tml)$">
            SetHandler "proxy:unix:/run/php/projet.nextcloud.localhost.sock|>
        </FilesMatch>
    </Directory>

    <IfModule mod_dav_fs.c>
        # Do not execute PHP files in webdav directory
        <Directory /var/www/nextcloud/webdav>
            <FilesMatch "\.php(p3?|tml)$">
                SetHandler None
            </FilesMatch>
        </Directory>
        DavLockDB /var/www/nextcloud/tmp/DavLock
    </IfModule>

    <IfModule mod_headers.c>
        Header always set X-Content-Type-Options "nosniff"
        Header always set X-Frame-Options "DENY"
        Header always set X-XSS-Protection "1; mode=block"
        Header always set Referrer-Policy "no-referrer"
    </IfModule>

    Protocols h2 h2c http/1.1
</VirtualHost>

^G Aide      ^O Écrire   ^W Chercher  ^K Couper   ^T Exécuter  ^C Emplacement
^X Quitter   ^R Lire fich.^V Remplacer  ^U Coller   ^J Justifier ^/ Aller ligne
```

La commande suivante active le VirtualHost pour le site "projet.nextcloud.localhost", ce qui permet à Apache de gérer les requêtes pour lui et de le rendre accessible aux utilisateurs :

```
root@debian:/var/www# a2ensite projet.nextcloud.localhost
```

Pour vérifier si le Virtual Host fonctionne, on commence par récupérer l'adresse IP de la machine. Ensuite, on ajoute une entrée dans le fichier hosts pour associer cette adresse à un nom de domaine virtuel. Cela permet d'accéder au site via ce nom dans un navigateur et de tester la configuration.

```
root@debian:/var/www# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:47:25:35 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.110.128/24 brd 192.168.110.255 scope global dynamic noprefixroute ens33
        valid_lft 1298sec preferred_lft 1298sec
    inet6 fe80::20c:29ff:fe47:2535/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

Maintenant on va forcer le pointage de cette adresse ip sur le nom projet.nextcloud.localhost, ce qui va nous permettre de faire fonctionner virtuellement ce nom de domaine sur cette machine. Ensuite il faudra faire la même chose sur les autres machines pour leur permettre d'accéder au site.

Cette façon de procéder est pratique dans notre cas où on reste en local, etc.

```
root@debian:/var/www# nano /etc/hosts
```

```
127.0.0.1    localhost
127.0.1.1    debian
192.168.110.128 projet.nextcloud.localhost
# The following lines are desirable for IPv6 capable hosts
::1          localhost ip6-localhost ip6-loopback
ff02::1      ip6-allnodes
ff02::2      ip6-allrouters
```

Maintenant on va créer un Fichier de test :

```
root@debian:/var/www# cd /var/www/nextcloud/web/
root@debian:/var/www/nextcloud/web# ls
root@debian:/var/www/nextcloud/web# nano test.php
```

```
GNU nano 7.2 test.php *
<?php
phpinfo();
```

PHP 8.2.26 - phpinfo()

projct.nextcloud.localhost/test.php

PHP Version 8.2.26

System	Linux debian 6.1.0-28-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.119-1 (2024
Build Date	Nov 25 2024 17:21:51
Build System	Linux
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.2/fpm
Loaded Configuration File	/etc/php/8.2/fpm/php.ini
Scan this dir for additional .ini files	/etc/php/8.2/fpm/conf.d
Additional .ini files parsed	/etc/php/8.2/fpm/conf.d/10-mysqld.ini, /etc/php/8.2/fpm/conf.d/10-opcache.ini, /etc/php/8.2/fpm/conf.d/10-pdo.ini, /etc/php/8.2/fpm/conf.d/15-xml.ini, /etc/php/8.2/fpm/conf.d/20-a
PHP API	20220829
PHP Extension	20220829
Zend Extension	420220829

Comme on le voit, notre serveur Web fonctionne correctement.

Etape 2 : Installation d'un serveur MariaDB

On commence par installer le serveur et client MariaDB pour la gestion des bases de données :

```
root@debian:/etc/apache2# apt install mariadb-server mariadb-client
```

On va maintenant configurer la première activation de MySQL

```
root@debian:/etc/apache2# mysql_secure_installation
```

```
Switch to unix_socket authentication [Y/n] y
```

```
Change the root password? [Y/n] n
```

```
Remove anonymous users? [Y/n] y
```

```
Disallow root login remotely? [Y/n] n
```

```
Remove test database and access to it? [Y/n] y
```

```
Reload privilege tables now? [Y/n] y
```

```
All done! If you've completed all of the above steps, your MariaDB  
installation should now be secure.
```

```
Thanks for using MariaDB!
```

Maintenant qu'il est installé, on va se connecter à MySQL

```
root@debian:/etc/apache2# Mysql -u root -p
-bash: Mysql : commande introuvable
root@debian:/etc/apache2# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 38
Server version: 10.11.6-MariaDB-0+deb12u1 Debian 12

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

Création de la base de données :

```
MariaDB [(none)]> CREATE DATABASE nextcloud;
Query OK, 1 row affected (0,001 sec)

MariaDB [(none)]>
```

On crée un utilisateur “nextcloud” via localhost, on lui attribue son mot de passe :

```
MariaDB [(none)]> CREATE USER 'nextclouduser'@'localhost' IDENTIFIED BY 'motdepasse';
Query OK, 0 rows affected (0,002 sec)
```

On lui accorde tous les privilèges sur la base de donnée :

```
MariaDB [(none)]> GRANT ALL PRIVILEGES ON nextcloud.* TO 'nextcloud'@'localhost';
Query OK, 0 rows affected (0,052 sec)
```

On applique les modifications des privilèges :

```
MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,023 sec)
```

On quitte maintenant MariaDB ;

```
MariaDB [(none)]> exit;
Bye
```

Notre serveur de base de données est maintenant configuré !

Etape 3 : Installation de Nextcloud

Maintenant que notre serveur Apache et notre SGBD sont prêts, nous allons pouvoir installer Nextcloud.

On se place tout d'abord à la racine de notre dossier Web :

```
root@debian:/tmp# cd /var/www/nextcloud/web/  
root@debian:/var/www/nextcloud/web#
```

Nous allons maintenant télécharger la dernière version de Nextcloud :

```
root@debian:/var/www/nextcloud/web# wget https://download.nextcloud.com/server/re  
leases/latest.zip
```

```
root@debian:/var/www/nextcloud/web# wget https://download.nextcloud.com/server/re  
leases/latest.zip  
--2025-01-04 13:51:26-- https://download.nextcloud.com/server/releases/latest.zi  
p  
Résolution de download.nextcloud.com (download.nextcloud.com)... 5.9.202.145, 2a01:  
4f8:210:21c8::145  
Connexion à download.nextcloud.com (download.nextcloud.com)|5.9.202.145|:443... con  
necté.  
requête HTTP transmise, en attente de la réponse... 200 OK  
Taille : 225508096 (215M) [application/zip]  
Sauvegarde en : « latest.zip »  
  
latest.zip          100%[=====>] 215,06M  16,4MB/s   ds 12s  
  
2025-01-04 13:51:40 (17,5 MB/s) – « latest.zip » sauvegardé [225508096/225508096]
```

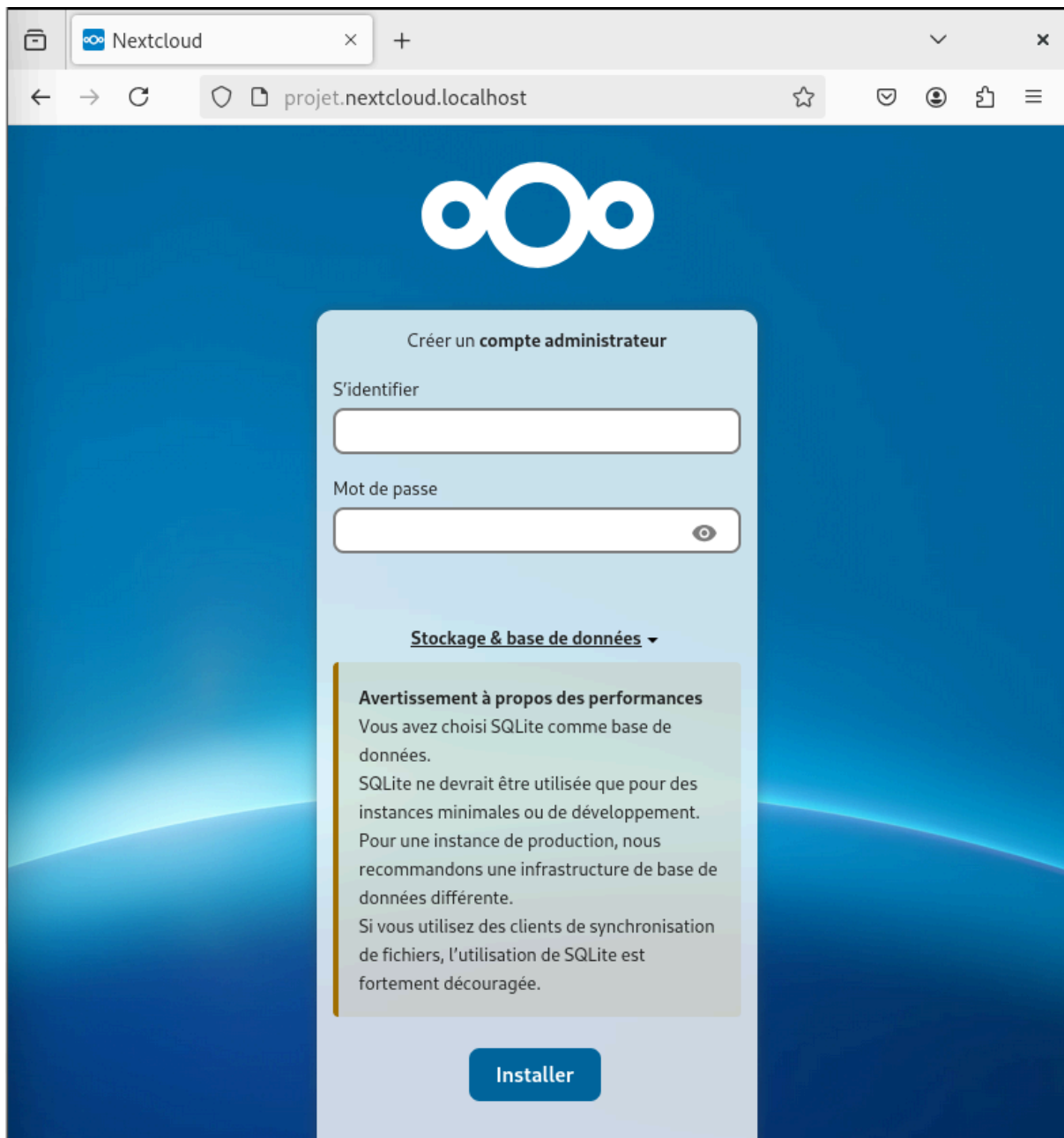
On dézippe l'archive :

```
root@debian:/var/www/nextcloud/web# unzip latest.zip
```


Les fichiers extraits de l'archive Nextcloud sont déplacés vers le répertoire racine du projet, avec suppression des fichiers inutiles. Ensuite, les permissions sont ajustées pour attribuer les fichiers à l'utilisateur et au groupe système dédiés (par exemple, web001), assurant une bonne organisation et la sécurité pour PHP-FPM.

```
root@debian:/var/www/nextcloud/web# rm latest.zip
root@debian:/var/www/nextcloud/web# mv nextcloud/* .
root@debian:/var/www/nextcloud/web# ls
3rdparty      console.php  index.php   ocs-provider  robots.txt
apps          COPYING     lib         package.json  status.php
AUTHORS       core        LICENSES   package-lock.json test.php
composer.json cron.php    nextcloud  public.php    themes
composer.lock dist        occ        remote.php    updater
config        index.html  ocs        resources     version.php
root@debian:/var/www/nextcloud/web# cd nextcloud/
root@debian:/var/www/nextcloud/web/nextcloud# ls -lart
total 24K
-rw-r--r--  1 root   root    271 12 déc.  11:00 .user.ini
drwxr-xr-x  2 root   root   4,0K 12 déc.  11:00 .reuse
-rw-r--r--  1 root   root   4,3K 12 déc.  11:00 .htaccess
drwxr-xr-x 15 web001 web001 4,0K  4 janv. 13:57 ..
drwxr-xr-x  3 root   root   4,0K  4 janv. 13:57 .
root@debian:/var/www/nextcloud/web/nextcloud# mv .user.ini ../
-bash: mv : commande introuvable
root@debian:/var/www/nextcloud/web/nextcloud# mv .htaccess ../
root@debian:/var/www/nextcloud/web/nextcloud# cd ..
root@debian:/var/www/nextcloud/web# rm -rf nextcloud/
root@debian:/var/www/nextcloud/web# cd ..
root@debian:/var/www/nextcloud# chown -R web001 web/
root@debian:/var/www/nextcloud#
```

On devrait maintenant pouvoir lancer l'installation depuis le navigateur, on se retrouve sur l'interface web de Nextcloud :



On voit qu'on reçoit un avertissement, il nous dit qu'on utilise SQLite, ce qui s'explique par le fait qu'il n'a pas encore reconnu notre base données MySQL :

The screenshot shows the 'Créer un compte administrateur' (Create an administrator account) screen in Nextcloud. The form includes fields for 'S'identifier' (username: 'projet') and 'Mot de passe' (password). Below these is a section titled 'Stockage & base de données' (Storage & database) with a dropdown for 'Répertoire des données' (data directory) set to '/var/www/nextcloud/web/data'. Under 'Configurer la base de données' (Configure the database), there are two tabs: 'SQLite' (selected) and 'MySQL/MariaDB'. Further down, there are fields for 'Compte de base de données' (database name: 'nextcloud'), 'Mot de passe de la base de données' (database password: 'nextcloud'), 'Nom de la base de données' (database name: 'nextcloud'), and 'Hôte de la base de données' (database host: 'localhost'). A note at the bottom states: 'Veuillez spécifier le numéro du port avec le nom de l'hôte (par exemple, localhost:5432)'. An 'Installer' button is at the bottom.

Créer un **compte administrateur**

S'identifier

projet

Mot de passe

●●●●●●●●

Stockage & base de données ▼

Répertoire des données

/var/www/nextcloud/web/data

Configurer la base de données

SQLite MySQL/MariaDB

Compte de base de données

nextcloud

Mot de passe de la base de données

nextcloud

Nom de la base de données

nextcloud

Hôte de la base de données

localhost

Veuillez spécifier le numéro du port avec le nom de l'hôte (par exemple, localhost:5432).

Installer

Nous voila connectés à Nextcloud !

