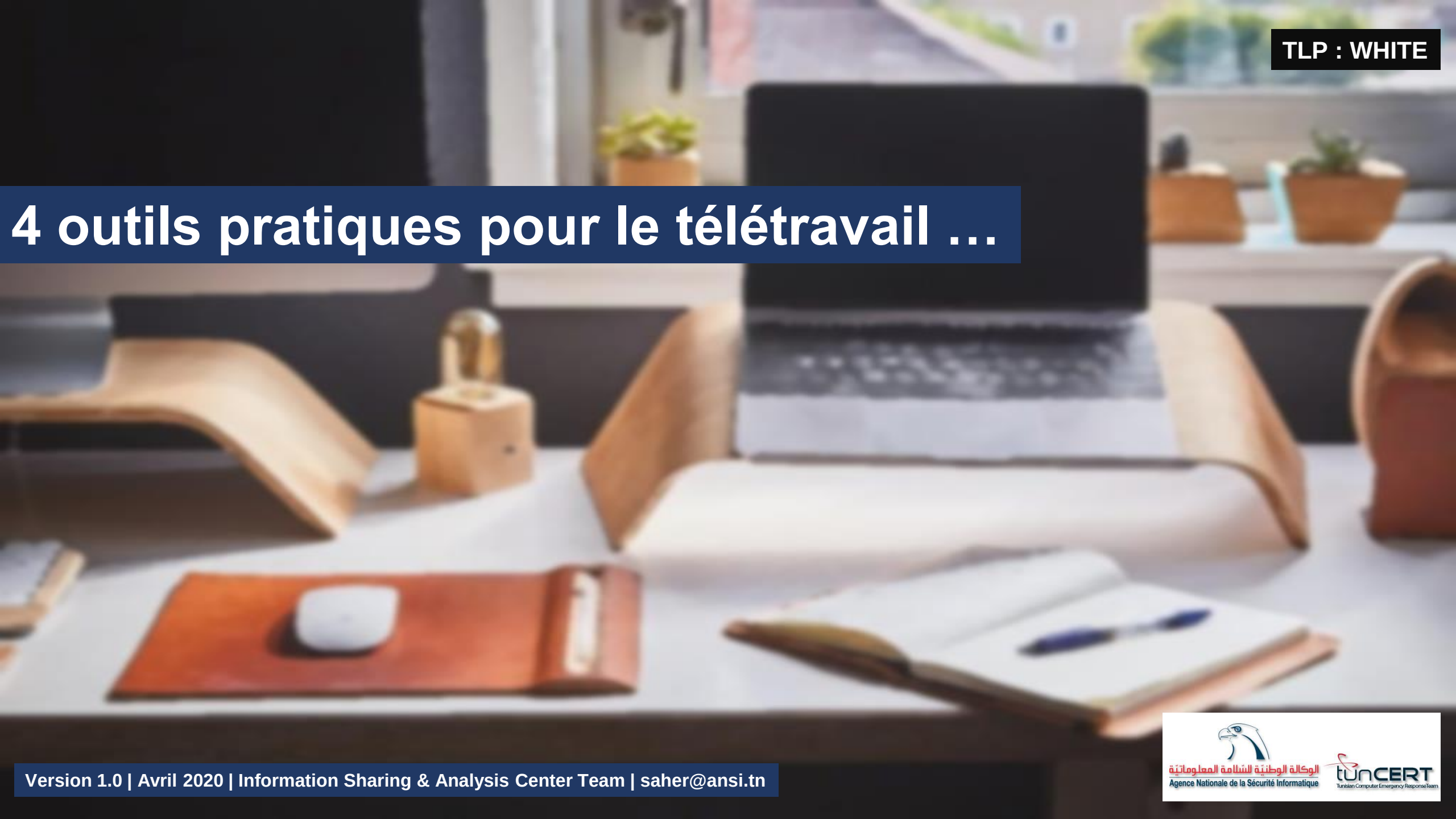


TLP : WHITE

4 outils pratiques pour le télétravail …

Version 1.0 | Avril 2020 | Information Sharing & Analysis Center Team | saher@ansi.tn

الوكالة الوطنيّة للسّلامة المعلوماتيّة
Agence Nationale de la Sécurité Informatique
tunCERT
Tunisian Computer Emergency ResponseTeam

Cadre Général du Plan de Continuité d'Activité PCA

Objectifs

Fréquemment, l'entreprise doit faire face à des risques, pouvant affecter son fonctionnement normal, qu'il s'agisse d'une grève dans une unité de production, d'un accident au sein d'une installation sensible ou encore une crise sanitaire (Coronavirus). Dans ces conditions, l'entreprise n'a d'autres choix que de mettre en place une organisation permettant de faire face à ces situations imprévues et d'en limiter les impacts potentiels, tant en termes de perte d'exploitation que de risque d'image.

Le Plan de Continuité d'Activité répond à cet objectif d'anticipation. Il fait partie intégrante de la politique de prévention des risques de l'entreprise, afin de garantir la continuité de ses activités lors de la survenance d'un sinistre.

Cadre Général du Plan de Continuité d'Activité PCA

Stratégies de continuité

Parmi les stratégies de continuité, on trouve le Télétravail :

L'une des mesures clés pour réduire la propagation du **Covid-19** est la **distanciation sociale**, ce qui pour de nombreuses entreprises signifie encourager le personnel à travailler à domicile.

Mais passer rapidement d'un environnement de bureau sécurisé à un travail à distance peut créer des risques de sécurité.

Cependant, il est tout à fait possible de limiter les risques des cyberattaques grâce à des mesures préventives :

- Utilisez un canal de communication sécurisée entre le poste de travail à distance et le réseau de votre entreprise par le biais d'un **VPN (Virtual Private Network)**
- Assurez-vous que l'anti-virus est en place et mis à jour.

Cadre Général du Plan de Continuité d'Activité PCA

Stratégies de continuité

- Sauvegarder vos données critiques dans des disques externes et des serveurs de backup protégés et isolés d'Internet.
- Mettre à jour régulièrement votre système d'exploitation, vos navigateurs Web et aussi votre solution antivirale.
- Créer périodiquement des points de restauration pour récupérer les fichiers système en cas d'infection.
- S'assurer que les accès à vos serveurs, vos équipements réseaux, vos ressources partagées et vos services en ligne (RDP, TELNET, SSH, FTP, SMB, NetBios, SMTP, POP3, etc. ...) soient limités, contrôlés et protégés avec des mots de passe robustes.
- Utiliser l'authentification à double facteur 2FA

4 outils indispensables pour le Télétravail

OpenVPN est un outil opensource qui sert à établir un tunnel sécurisé entre un client et un serveur. Il peut fonctionner en TCP ou UDP. Le serveur OpenVPN peut être installé sur Windows et Linux. Il existe un client sur presque toutes les plateformes.

Nextcloud est une solution de partage de document et de collaboration pour les employés. Il est accessible à partir d'un simple navigateur ou à partir de clients spécifiques à chaque OS (desktop et mobile).

Mattermost est l'alternative Opensource de Slack. Le but de ce projet est de fournir un moyen de collaboration sécurisé et flexible. Il fournit des fonctionnalités avancées destinées aux entreprises du secteur IT (intégration avec Github, Jira, Docker, Google Tools, Jenkins, Redmine, etc...)

BigBlueButton est une solution de conférence web qui permet d'organiser des réunions et des formations en ligne (Classes Virtuelles) mais également de les enregistrer pour créer des ressources pédagogiques.

Installation et Configuration d'un Tunnel VPN

Type VPN

– IPSec agit au niveau de la couche IP:

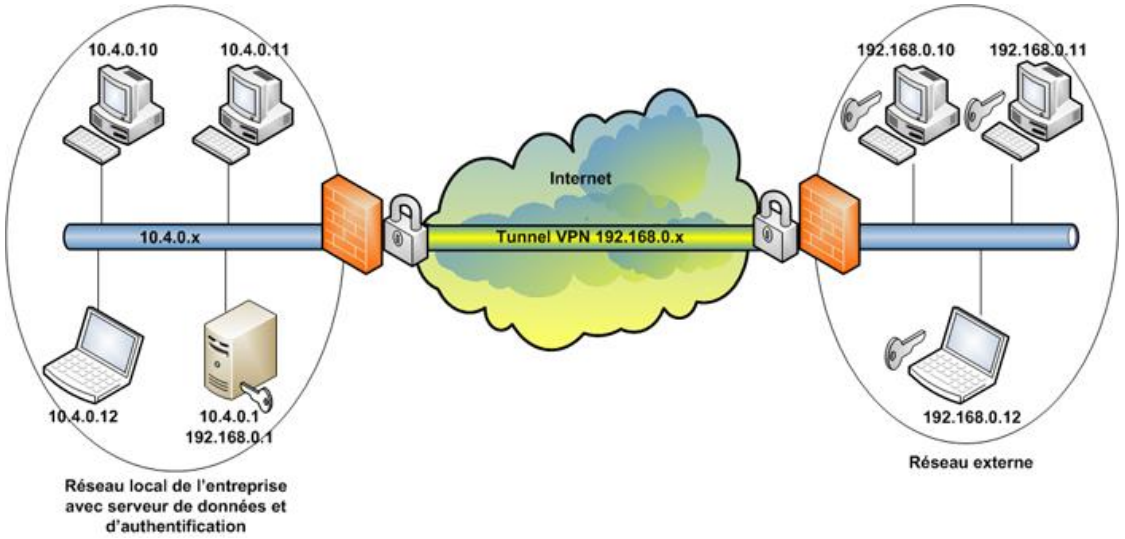
- Protection unique pour l'ensemble des applications
- Protège de bout en bout
- Protège les données pair à pair
- L'ensemble des équipements utilisant le tunnel bénéficient de ces avantages

– VPN SSL dans le cas de ***OpenVPN***, agit sur les couches réseau, transport, session, présentation et application:

- Authentification et chiffrement par SSL
- Configuration à réaliser sur chacun des clients
- Passe facilement les NAT

OpenVPN

- Il existe plusieurs configurations possibles avec ***OpenVPN***:
- La première consiste à utiliser des clés statiques, simple à mettre en œuvre mais peu adaptée dans une architecture avec beaucoup de clients et peu sécurisée. En effet, l'interception de la clé partagée va dévoiler tous les échanges y compris ceux déjà passés.
- La seconde, plus sécurisée, consiste à utiliser des certificats. C'est celle-ci que nous allons mettre en place. Nous monterons un tunnel en UDP sur le port 1194 (port par défaut).



Installation et Configuration

- **Serveur**

- Avant de commencer, mettez à jour votre serveur:

sudo apt-get update

sudo apt-get upgrade

- Installez les paquets nécessaires:

sudo apt-get install openvpn easy-rsa

- Le serveur est installé, nous allons maintenant le configurer. Commençons par générer les certificats et clés nécessaires. Nous allons tout mettre dans le même dossier, lancez les deux commandes suivantes afin de le créer et d'aller dans ce dossier:

make-cadir certificats cd certificats

Installation et Configuration

- A partir de cette étape, lancez toutes les commandes dans le dossier que vous venez de créer « certificats ». Nous allons ensuite modifier les variables:

sudo nano vars

- Afin de définir précisément le chemin du fichier openssl.cnf. Remplacez la ligne:

export KEY_CONFIG=`$EASY_RSA/whichopensslcnf $EASY_RSA`
Par

export KEY_CONFIG="$EASY_RSA/openssl-1.0.0.cnf"

Toujours dans le fichier vars, vous pouvez modifier les valeurs qui vous sont proposées par défaut lors de la création du certificat, en remplaçant les valeurs entre « »:

Installation et Configuration

- A partir de cette étape, lancez toutes les commandes dans le dossier que vous venez de créer « certificats ». Nous allons ensuite modifier les variables:

sudo nano vars

- Afin de définir précisément le chemin du fichier openssl.cnf. Remplacez la ligne:

export KEY_CONFIG=`$EASY_RSA/whichopensslcnf $EASY_RSA`
Par

export KEY_CONFIG="$EASY_RSA/openssl-1.0.0.cnf"

Toujours dans le fichier vars, vous pouvez modifier les valeurs qui vous sont proposées par défaut lors de la création du certificat, en remplaçant les valeurs entre « »:

Installation et Configuration

export KEY_COUNTRY=“TN"

export KEY_PROVINCE=“TN"

export KEY_CITY=“TUNIS"

export KEY_ORG=“ANSI"

export KEY_EMAIL=“saher@ansi.tn"

Export KEY_OU=“ISAC“

- Pour prendre en compte ces nouvelles variables, entrez la commande ci-dessous:

source vars

- Notre environnement est prêt, nous allons pouvoir créer notre autorité de certification. Utilisez les scripts présents dans le dossier certificats:

./clean-all && ./build-ca

Remplissez les champs ou laissez par défaut si vous aviez déjà modifié les valeurs dans le fichier vars.

Installation et Configuration

- Notre autorité de certification étant créée, nous allons générer le certificat et la clé du serveur OpenVPN. Lancez la commande suivante:

./build-key-server server

- Nous allons générer les paramètres Diffie-Hellman, cette étape peut prendre quelques minutes:

./build-dh

- En plus d'utiliser un certificat, nous allons également générer une clé TLS pour avoir encore plus de sécurité. Utilisez la commande ci-dessous:

openvpn --genkey --secret keys/ta.key

Installation et Configuration

- Copiez les fichiers qui sont dans /certificats/keys dans le dossier /etc/openvpn afin qu'ils soient pris en compte:

sudo cp keys/{server.crt,server.key,ca.crt,dh2048.pem,ta.key} /etc/openvpn

- Pour la configuration de OpenVPN, un exemple du fichier server.conf est situé ici : /usr/share/doc/openvpn/examples/sample-config-files/server.conf.gz. Nous allons le décompresser et le placer dans /etc/openvpn:

gzip -d -c /usr/share/doc/openvpn/examples/sample-config-files/server.conf.gz | sudo tee /etc/openvpn/server.conf

- Par défaut, seul le trafic entre votre serveur et votre client va passer dans le tunnel (le trafic internet ne passera pas dans celui-ci par exemple). Dans le fichier server.conf, nous allons dé-commenter la ligne qui concerne ce point afin que tout le trafic passe dans le tunnel:

sudo nano /etc/openvpn/server.conf

Installation et Configuration

- Dé-commentez la ligne:

push "redirect-gateway def1 bypass-dhcp"

- Contenu du fichier de configuration du serveur ***OpenVPN*** (server.conf):

```
port 1194
proto udp
dev tun
ca ca.crt
cert server.crt
key server.key # This file should be kept secret
dh dh2048.pem #vous allez trouver dh1024.pem
server 10.8.0.0 255.255.255.0
ifconfig-pool-persist /var/log/openvpn/ipp.txt
push "redirect-gateway def1 bypass-dhcp"
keepalive 10 120 tls-auth ta.key 0 # This file is secret
cipher AES-256-CBC
persist-key
persist-tun
status /var/log/openvpn/openvpn-status.log
verb 3
explicit-exit-notify 1
```

- Démarrez le service OpenVPN:

sudo systemctl start openvpn

- Vérifiez qu'il est bien actif:

sudo systemctl status openvpn

Installation et Configuration

- Nous allons maintenant passer à la configuration des clients. Nous allons créer le certificat du client pour le serveur ainsi que le certificat et la clé pour le client.

- Utilisez la commande ci dessous (vérifiez que vous êtes toujours dans le répertoire certificats):

source vars && ./build-key client

- Créez un répertoire pour le client dans /etc/openvpn et copiez le certificat du client, la clé du client, le certificat du serveur et la clé TLS dedans:

sudo mkdir /etc/openvpn/client1

sudo cp keys/{client.crt,client.key,ca.crt,ta.key} /etc/openvpn/client1 && cd /etc/openvpn/client1

Installation et Configuration

- Nous allons créer le fichier de configuration du client. Comme pour le serveur nous allons utiliser un modèle déjà présent sur le serveur:

sudo cp /usr/share/doc/openvpn/examples/sample-config-files/client.conf client.ovpn

- Éditez le fichier:

sudo nano client.ovpn

- Sur la ligne suivante, remplacez « my-server » par l'IP de votre serveur ***OpenVPN***, laissez 1194 si vous avez laissé le port par défaut sinon modifiez-le:

remote my-server-1 1194

Installation et Configuration

- Le contenu de client.ovpn sera le suivant:

```
client
dev tun
proto udp
remote IP_SRV_OPENVPN 1194
 resolv-retry infinite
 nobind persist-key
persist-tun
ca ca.crt
cert client.crt
key client.key
remote-cert-tls server
tls-auth ta.key 1
cipher AES-256-CBC
verb 3
```

Installation et Configuration

- Nous allons créer un zip contenant les informations devant être sur le poste client. Le fichier zip contiendra:

  - **ca.crt : certificat du serveur *OpenVPN***
  - **client.crt : certificat du client**
  - **client.key : clé du client**
  - **ta.key : clé TLS**
  - **client.ovpn : fichier configuration vpn**

- Dans le répertoire du client, faites un zip du dossier:

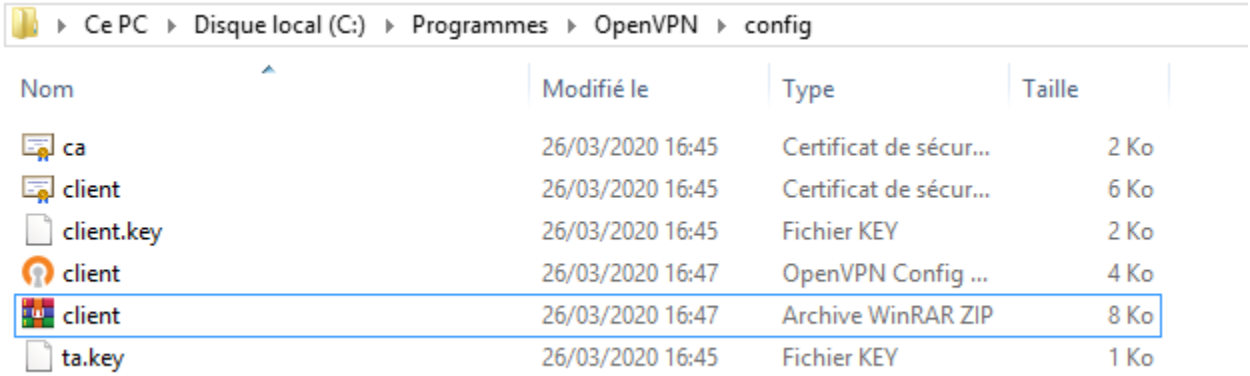
sudo zip client.zip *.*

- C'est ce fichier que nous allons mettre sur le client afin qu'il puisse se connecter au serveur ***OpenVPN.***

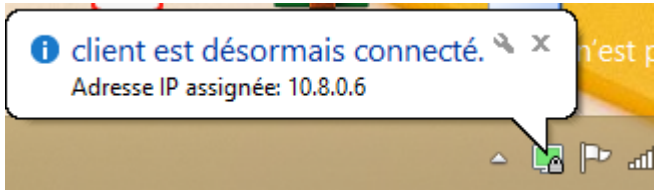
Installation et Configuration

- Client

- Dézipper le fichier sous « C:\Program Files\OpenVPN\config »



- Ouvrir OpenVPN et connectez vous avec le client



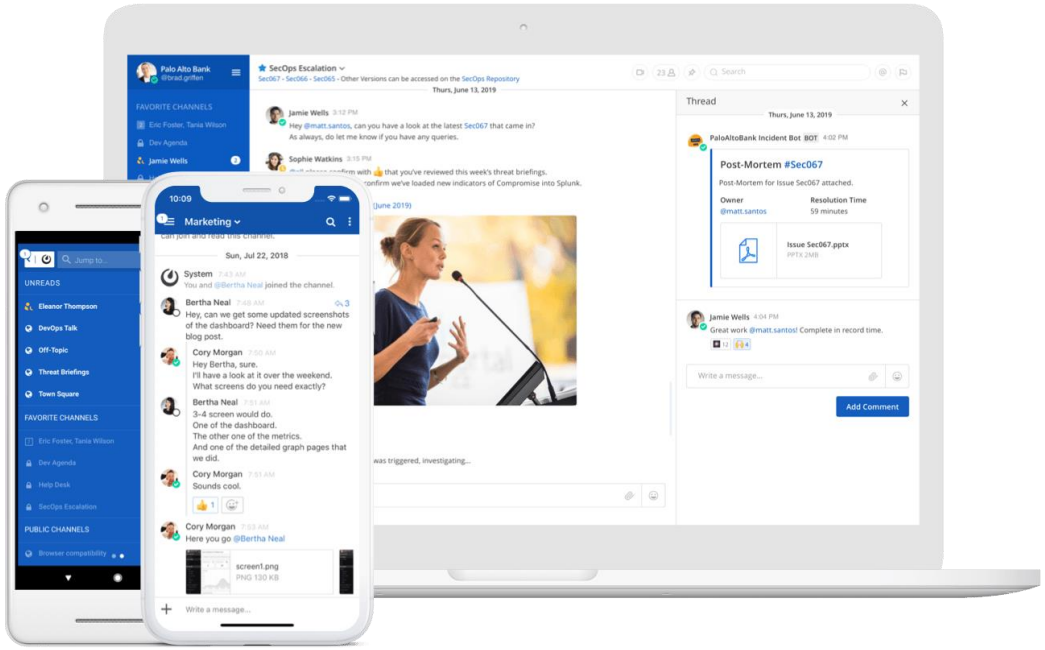
Installation et Configuration d’un serveur Mattermost

Présentation de Mattermost

– Mattermost est un service de discussion instantanée open source en auto-hébergement. Il est conçu comme un chat interne pour les organisations et les entreprises, et il est présenté comme une alternative à Slack.

– Fonctionnalités proposées:

- Classement des discussions en équipes (Teams) et en canaux (Channels)
- Applications natives pour les différentes plateformes
- Recherche dans l'historique des messages
- Plugins pour l'intégration de outils de travails existants
- Traduction en temps réel
- Etc..



Prérequis serveur Mattermost

- Hardware
  - RAM : 2GB
  - Disque : selon le nombre des utilisateurs, et leur activité (upload fichiers).
    Prévoir au moins 20 GB pour une petite équipe et une utilisation modérée .

- Software
  - Système d'exploitation: Ubuntu 16.04, Ubuntu 18.04, Debian Buster, CentOS 6+, CentOS 7+, RedHat Enterprise Linux 6+, RedHat Enterprise Linux 7+, Oracle Linux 6+, Oracle Linux 7+
  - SGBD: MySQL (5.6, 5.7, 8), PostgreSQL 9.4+

Installation serveur Mattermost

Plan de l'installation:

- Mettre à jour l'OS
- Installation du SGBD
- Installation Mattermost
- Configuration

Installation: Mettre à jour l'OS

Sur un serveur de la famille Ubuntu / Debian:

- `sudo apt-get update`
- `sudo apt-get upgrade`

Sur un serveur de la famille RHEL / CentOS:

- `sudo yum update`
- `sudo yum upgrade`

Installation: SGBD

Installation MySQL:

```
sudo apt-get install mysql-server
```

Connexion en tant que root:

```
mysql -u root -p
```

Création d'un utilisateur pour Mattermost

```
mysql> create user 'mmuser'@'%' identified by 'mmuser-password';
```

Création de la base de données Mattermost

```
mysql> create database mattermost;
```

Affectation des droits nécessaires

```
mysql> grant all privileges on mattermost.* to 'mmuser'@'%';
```

Sortie

```
mysql> exit
```

Installation: Mattermost

Téléchargement de la dernière version de Mattermost

```
wget https://releases.mattermost.com/X.X.X/mattermost-X.X.X-linux-amd64.tar.gz
```

Vous pouvez trouver le lien ici : https://about.mattermost.com/download/

Extraire les fichiers

```
tar -xvzf mattermost*.gz
```

Déplacer les fichiers vers /opt

```
sudo mv mattermost /opt
```

Créer un dossier pour le stockage des fichiers utilisateurs

```
sudo mkdir /opt/mattermost/data
```

Créer un utilisateur système Mattermost, et affecter les permissions

```
sudo useradd --system --user-group mattermost
sudo chown -R mattermost:mattermost /opt/mattermost
sudo chmod -R g+w /opt/mattermost
```

Installation: Configuration

Configurer les paramètres de connexion SGBD

- Ouvrir le fichier /opt/mattermost/config/config.json
- Mettre `"DriverName"` à `"mysql"`
- Mettre `"DataSource"` à `"mmuser:<mmuser-password>@tcp(<host-name-or-IP>:3306)/mattermost?charset=utf8mb4,utf8&readTimeout=30s&writeTimeout=30s"`
- Mettre `"SiteURL"` à l'URL du serveur (par ex: `"https://mattermost.entreprise.tn"`)

Installation: Configuration

Création d'un fichier pour le démarrage/arrêt automatisé

```
sudo nano /lib/systemd/system/mattermost.service
```

avec le contenu suivant:

```
[Unit]
Description=Mattermost
After=network.target
After=mysql.service
Requires=mysql.service

[Service]
Type=notify
ExecStart=/opt/mattermost/bin/mattermost
TimeoutStartSec=3600
Restart=always
RestartSec=10
WorkingDirectory=/opt/mattermost
User=mattermost
Group=mattermost
LimitNOFILE=49152

[Install]
WantedBy=mysql.service
```

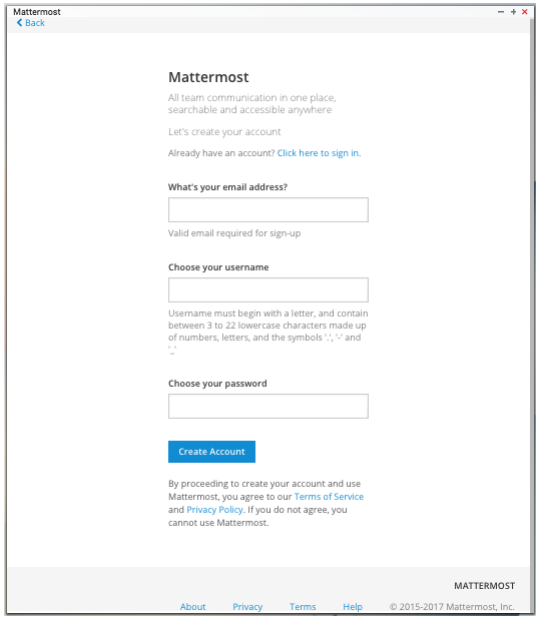
Installation: Test

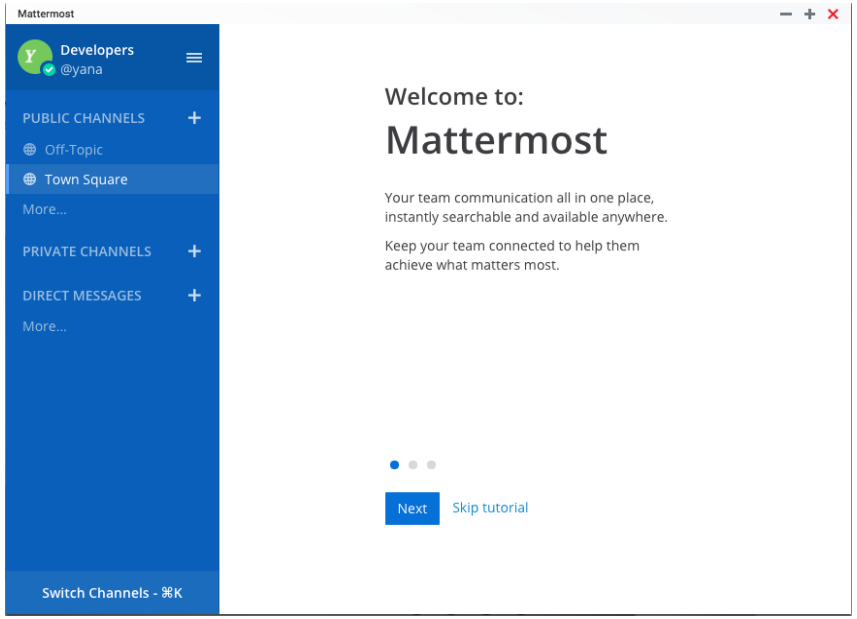
Démarrer le service avec la commande suivante:

```
sudo systemctl start mattermost.service
```

Par la suite, ouvrir l'URL suivante : http://mattermost.entreprise.tn:8065

Suivre les étapes pour la création du compte Admin, et la première équipe:





BigBlueButton

Installation et configuration d'une solution de conférence Web

Installation

- BigBlueButton est une solution de conférence web qui permet d'organiser des réunions et des formations en ligne (Classes Virtuelles) mais également de les enregistrer pour créer des ressources pédagogiques.
- Il vous suffit de lancer la commande suivante sur un serveur Ubuntu 16.04

Installation avec SSL/TLS (nécessite ip public + FQDN)

wget -qO- https://ubuntu.bigbluebutton.org/bbb-install.sh | bash -s -- -v xenial-220 -s bbb.example.com -e info@example.com

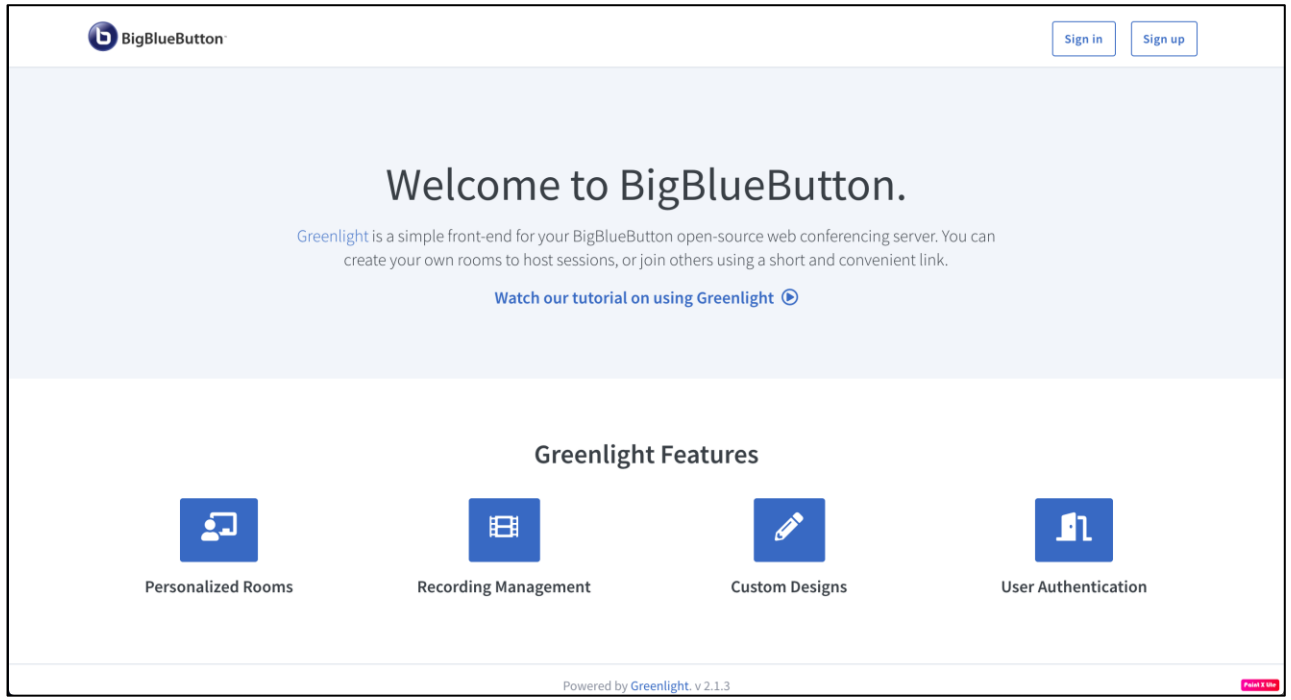
Cette commande permet d'automatiser les étapes de ce lien

http://docs.bigbluebutton.org/2.2/install.html

Installation

- La deuxième étape consiste à installer l'interface graphique Greenlight

wget -qO- https://ubuntu.bigbluebutton.org/bbb-install.sh | bash -s -- -v xenial-220 -s bbb.example.com -e info@example.com -g



https://github.com/bigbluebutton/bbb-install

Installation d'une solution de travail collaboratif

NextCloud

– Plusieurs entreprises sont soucieux de la sécurité de leurs données dans le cloud. Heureusement, des solutions sont disponibles pour cela, comme NextCloud.

– NextCloud est une solution de stockage et de partage de fichiers en ligne. C'est une alternative open source à OwnCloud. La principale différence entre NextCloud et OwnCloud est que NextCloud est complètement open source. De même, NextCloud élimine votre besoin d'utiliser un logiciel d'hébergement cloud tiers comme Dropbox pour stocker vos documents.

- NextCloud propose également de nombreuses fonctionnalités avancées qui le permettent d'être intégré facilement dans une entreprise.

Caractéristiques

- Permettre de gérer les utilisateurs et les groupes à l'aide de LDAP
- Permettre d’accéder, de synchroniser et de partager vos données existantes sur Dropbox, FTP et NAS.
- Supporte une sécurité avancée : authentification à deux facteurs.
- Partager des fichiers avec d'autres utilisateurs, créer et envoyer des liens publics protégés par mot de passe.
- Supporte la visioconférence.
- Envoi de notification lorsque quelqu'un sur le serveur a partagé des fichiers directement avec vous.

Installation et Configuration

- **Prérequis**

- OS : Debian 9
- Une adresse IP statique de votre serveur: @IP. Vous pouvez utiliser un nom de domaine.
- Pour configurer Nextcloud, on doit avoir le serveur LAMP sur votre système Debian 9. Si l'on a, on peut ignorer cette étape.

Installation et Configuration

- **Mise à jour du système**

- Mettez à jour votre système

$ su (changer à root)

#apt-get update -y

#apt-get upgrade -y

- Redémarrez votre système

#reboot

- **Installation de LAMP**

- NextCloud fonctionne avec Apache, MySQL et PHP. Vous devrez donc installer tous ces composants sur votre système.

Installation et Configuration

- Installez le serveur Apache et MariaDB.

#apt-get install apache2 mariadb-server apt-transport-https –y

- Une fois l'installation terminée, démarrez le service Apache et MariaDB et activez-les.

 # systemctl start apache2 && systemctl enable apache2 && systemctl start mysql && systemctl enable mariadb

- Ajouter le repository « Ondrej Debian »

#wget -q https://packages.sury.org/php/apt.gpg -O- | apt-key add -

#echo "deb https://packages.sury.org/php/ stretch main" | tee /etc/apt/sources.list.d/ondrej.list

#apt-get update

- Installez PHP et d'autres modules requis sur votre système.

Installation et Configuration

#apt-get install libapache2-mod-php php php-xml php-curl php-gd php-cgi php-cli php-zip php-mysql php-mbstring wget unzip –y

- Vérifiez votre version PHP

#php -version

```
root@[redacted]~# php -version
PHP 7.4.4 (cli) (built: Mar 20 2020 14:24:19) ( NTS )
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies
    with Zend OPcache v7.4.4, Copyright (c), by Zend Technologies
```

- Ensuite, il est nécessaire de modifier certaines options PHP. Pour ce faire, modifiez le fichier de configuration php.ini

#nano /etc/php/7.4/apache2/php .ini

Installation et Configuration


```
memory_limit = 512M
upload_max_filesize = 200M
max_execution_time = 360
post_max_size = 200M
```

- Redémarrez Apache

 # systemctl restart apache2

- **Configuration de MariaDB**

- Par défaut, l'installation de MariaDB n'est pas sécurisée, donc on doit la sécuriser.

Installation et Configuration

#mysql_secure_installation

- Il vous sera posé des questions sur la configuration de MariaDB. On peut répondre comme suit.

```
//
Enter current password for root (enter for none):
OK, successfully used password, moving on...
//
Remove anonymous users? [Y/n] y
Disallow root login remotely? [Y/n] y     (Ici , on peut mettre Non si on veut activer l'accès à distance avec le root)
Remove test database and access to it? [Y/n] y
Reload privilege tables now? [Y/n] y
```

Installation et Configuration

- Connectez à MariaDB

#mysql -u root -p

- Créez une base de donnée nommée « nextcloud » et un utilisateur « nextcloud » tout en attribuant à lui un mot de passe et tous les privilèges sur cette base.

> CREATE DATABASE nextcloud;

> CREATE USER 'nextcloud'@'localhost' IDENTIFIED BY '123';

> GRANT ALL PRIVILEGES ON nextcloud.* TO 'nextcloud'@'localhost';

> quit ;

- Remplacez « 123 » par votre mot de passe.

- **Installation de NextCloud**

Installation et Configuration

- Premièrement, créez un répertoire pour l'installation de NextCloud.

#mkdir /var/www/html/nextcloud

#chown -R www-data:www-data /var/www/html/nextcloud/

#chmod -R 755 /var/www/html/nextcloud/

- Deuxièmement, créez un répertoire où NextCloud va stocker les fichiers téléchargés.

#mkdir -p /var/nextcloud/data

#chown -R www-data:www-data /var/nextcloud/data

#chmod 755 /var/nextcloud/data

- Troisièmement, configurerz un fichier des hôtes virtuels pour NextCloud.

#nano /etc/apache2/sites-available/nextcloud.conf

Installation et Configuration


```
<VirtualHost *:80>
DocumentRoot "/var/www/html/nextcloud"
ServerName @IP (On peut mettre un nom de domaine ici)
<Directory "/var/www/html/nextcloud/">
        Options MultiViews FollowSymlinks
        AllowOverride All
        Order allow,deny
        Allow from all
</Directory>
TransferLog /var/log/apache2/nextcloud_access.log
ErrorLog /var/log/apache2/nextcloud_error.log
</VirtualHost>
```

Installation et Configuration

- Maintenant, activez ce fichier et redémarrez apache.

#a2ensite nextcloud && systemctl reload apache2

- Finalement, télécharger NextCloud depuis leur site officiel « https://download.nextcloud.com/server/releases/ »

- On va installer la dernière version et la décompresser.

#wget https://download.nextcloud.com/server/releases/latest.zip

#unzip latest.zip

- Déplacez le dossier et modifiez les permissions sur lui.

#mv nextcloud /var/www/html/

#chown -R www-data:www-data /var/www/html/nextcloud

Installation et Configuration

- **Accès à l'interface web de NextCloud**

- Ouvrez l'url suivante sur votre navigateur :

« **http://@IP/setup-nextcloud.php** ».

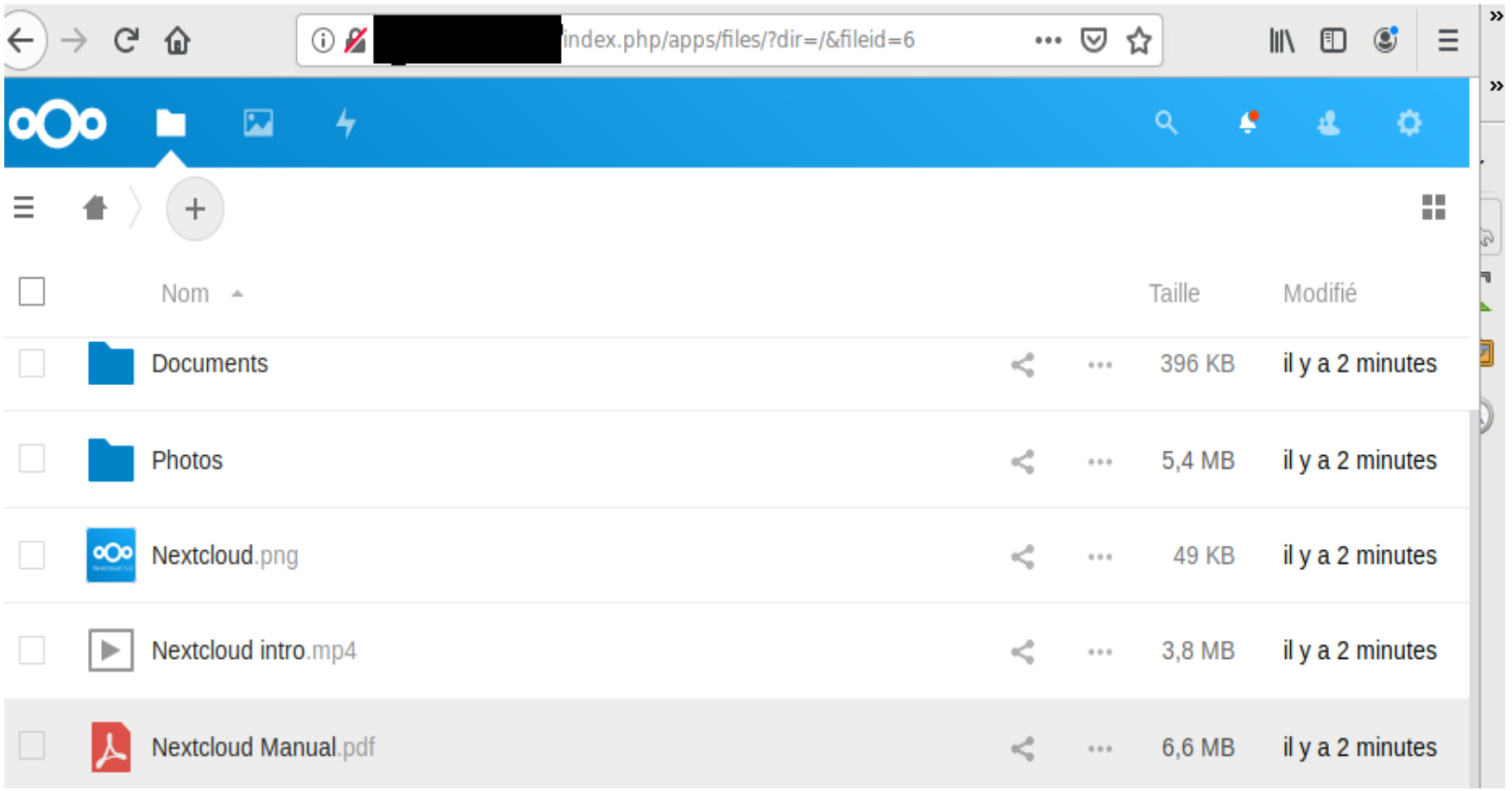
Installation et Configuration

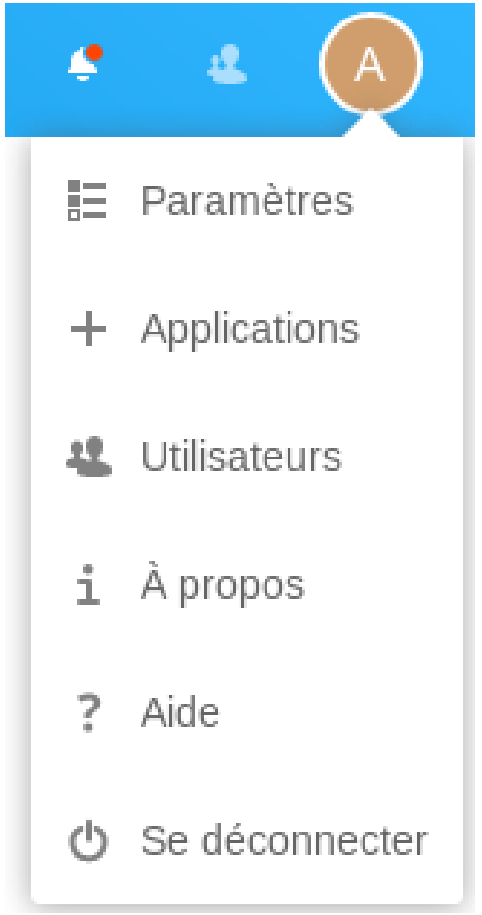
- Ajoutez les identifiants pour créer un compte d'administrateur et fournissez les informations relatives à la base de données (nom d'utilisateur de la base, mot de passe, nom de la base).

On doit avoir le Dashboard de NextCloud comme celle-ci.

Installation et Configuration

- Après avoir terminé la configuration, on peut créer des utilisateurs et les assigner à des groupes, partager des fichiers avec les autres,...





saher@ansi.tn