


```
chown -R www-data /var/www/ogis-server
```

Créons un fichier /etc/apache2/sites-avaible/ogis-server.conf. Une bonne pratique pour QGIS Server et de restreindre l'accès à localhost uniquement (seul le client web Lizmap accèdera à QGIS Server) ; c'est pourquoi on peut utiliser le port 80, sans certificat.

```
<VirtualHost 127.0.0.1:80>
  ServerAdmin webmaster@localhost
  ServerName ows.local

  #DocumentRoot /var/www/html

  # Apache logs (different than QGIS Server log)
  ErrorLog $(APACHE_LOG_DIR)/ogis-server.error.log
  CustomLog $(APACHE_LOG_DIR)/ogis-server.access.log combined

  # Longer timeout for WPS... default = 40
  FcgidIOTimeout 120

  FcgidInitialEnv LC_ALL "fr_FR.UTF-8"
  FcgidInitialEnv PYTHONIOENCODING UTF-8
  FcgidInitialEnv LANG "fr_FR.UTF-8"

  # QGIS log
  FcgidInitialEnv QGIS_SERVER_LOG_STDERR 0
  FcgidInitialEnv QGIS_SERVER_LOG_LEVEL 0

  FcgidInitialEnv QGIS_PLUGINPATH "/var/www/ogis-server/plugins"
  FcgidInitialEnv QGIS_SERVER_LIZMAP_REVEAL_SETTINGS True

  # default QGIS project
  # SetEnv QGIS_PROJECT_FILE /var/www/ogis-server/projects/fr_Physique.qgz
  SetEnv QGIS_PROJECT_FILE /var/www/ogis-server/projects/fr_Physique.qgz

  # QGIS_AUTH_DB_DIR_PATH must lead to a directory writeable by the Server's FCGI process user
  FcgidInitialEnv QGIS_AUTH_DB_DIR_PATH "/var/www/ogis-server/db/"
  FcgidInitialEnv QGIS_AUTH_PASSWORD_FILE "/var/www/ogis-server/db/ogis-auth.db"

  # Set pg access via pg_service file
  SetEnv PGSERVICEFILE /home/sist/.pg_service.conf
  FcgidInitialEnv PGPASSFILE "/home/sist/.pgpass"

  # if ogis-server is installed from packages in debian based distros this is usually /usr/lib/cgi-bin/
  # run "locate qgis_mapserv.fcgi" if you don't know where qgis_mapserv.fcgi is
  ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
  <Directory "/usr/lib/cgi-bin/">
    AllowOverride None
    Options +ExecCGI -MultiViews -SymLinksIfOwnerMatch
    Require all granted
  </Directory>

  <IfModule mod_fcgid.c>
    FcgidMaxRequestLen 26214400
    FcgidConnectTimeout 60
    # Needed for QGIS HelloServer plugin HTTP BASIC auth
    RewriteEngine on
    RewriteCond %{HTTP:Authorization} .
    RewriteRule .* - [E=HTTP_AUTHORIZATION:%{HTTP:Authorization}]
  </IfModule>

</VirtualHost>
```

Activez le virtual host et relancez Apache2

```
sudo a2ensite ogis-server.conf
sudo systemctl restart apache2
```

Nous pouvons maintenant tester QGIS Server avec cette requête GetCapabilities :

[http://ows.local/cgi-bin/qgis_mapserv.fcgi?](http://ows.local/cgi-bin/qgis_mapserv.fcgi?SERVICE=WMS&VERSION=1.3.0&REQUEST=GetCapabilities)

[SERVICE=WMS&VERSION=1.3.0&REQUEST=GetCapabilities](http://ows.local/cgi-bin/qgis_mapserv.fcgi?SERVICE=WMS&VERSION=1.3.0&REQUEST=GetCapabilities)

Voir aussi https://docs.qgis.org/3.28/fr/docs/server_manual/getting_started.html#installation-on-debian-based-systems

Installation de PHP7 et PHP-FPM

Nous allons passer à l'installation du client web Lizmap, qui requiert PHP version 7 ou 8 (sur cette machine : PHP 8.1). 3Liz recommande l'utilisation de PHP-FPM pour améliorer la performance (PHP en tant que service distinct d'Apache).

```
sudo install php8.1-fpm
sudo apt install php8.1-curl php8.1-gd php8.1-imagick php8.1-mcrypt php8.1-openssl php8.1-soap php8.1-xml
sudo apt install curl php8.1-sqlite3 php8.1-gd php8.1-xml php8.1-curl php8.1-psql
sudo a2dismod php8.1
sudo a2dismod mpm_prefork
sudo a2enmod mpm_event proxy_fcgi setenvif
sudo a2enconf php8.1-fpm
sudo systemctl status php8.1-fpm
```

Installation du client web Lizmap (3.6)

Tip

Voir aussi <https://github.com/3liz/lizmap-web-client/blob/master/INSTALL.md>

Installation du client web

Télécharger et dézipper le client web Lizmap : <https://github.com/3liz/lizmap-web-client/releases> (dernière release stable : 3.6.5, à la date du 28/09/2023)

Le copier dans `/var/www/`

```
unzip lizmap-web-client-3.6.5.zip
sudo cp -r lizmap-web-client-3.6.5 /var/www/lizmap-web-client
```

Copier les fichiers de configuration et lancer l'installation

```
cd /var/www/lizmap-web-client/lizmap/var/config
sudo cp lizmapConfig.ini.php.dist lizmapConfig.ini.php
sudo cp localConfig.ini.php.dist localConfig.ini.php
sudo cp profiles.ini.php.dist profiles.ini.php
cd /var/www/lizmap-web-client
sudo lizmap/install/set_rights.sh www-data www-data
sudo php lizmap/install/installer.php
```

Vous devez voir à l'écran :

```
Installation is starting
All modules dependencies are ok
Installation starts
Module jelix installed
Module jcommunity installed
Module jaci2 installed
Module jaci2db installed
Module jaci2db_admin installed
Module jauthdb_admin installed
Module master_admin installed
Module admin installed
Module lizmap installed
Module view installed
Module action installed
Module dynamicLayers installed
Module proxy4php installed
Module dataviz installed
Module filter installed
All modules are installed or upgraded
```

```
Installation is finished
```

Relancer la définition des droits dans un terminal :

```
cd /var/www/lizmap-web-client
sudo lizmap/install/set_rights.sh www-data www-data
```

Création d'un hôte virtuel

Créons un hôte virtuel sur `lizmap.local` pour la machine virtuelle de TP.

Créer un nouveau fichier `/etc/apache2/sites-available/lizmap.conf` :

```
<VirtualHost *:80>
    ServerName lizmap.local
    DocumentRoot /var/www/lizmap-web-client/lizmap/www
    <Directory /var/www/lizmap-web-client>
        AllowOverride All
        Require all granted
    </Directory>
    ErrorLog ${APACHE_LOG_DIR}/error_lizmap.log
    CustomLog ${APACHE_LOG_DIR}/access_lizmap.log combined
    #Include conf-available/serve-cgi-bin.conf
</VirtualHost>
```

Activer le site et redémarrer apache :

```
sudo a2ensite lizmap.conf
sudo systemctl restart apache2
```

Vous pouvez maintenant vous rendre sur le site <http://lizmap.local/>

Changer le mot de passe en : `admin` / `anf2023`

Pour passer le site en https, nous générons un certificat autosigné et configurons le virtual host sur le port 443 dans `/etc/apache2/sites-available/lizmap.conf`,

Données de test dans Lizmap

Nous créons 2 répertoires qui contiendra les cartes Lizmap. Le premier `public` sera accessible à tous, y compris les utilisateurs anonymes. Le second `privé` ne sera accessible qu'aux administrateurs et utilisateurs authentifiés.

```
cd ~
sudo mkdir -p data/mon_travail/3_lizmap/public
sudo mkdir -p data/mon_travail/3_lizmap/privé
```

Puis nous copions dans `public` le projet QGIS qui servira à tester le bon fonctionnement Lizmap (sur la machine virtuelle, les fichiers : `fra_physical.qgs` et `fra_physical.qgs.cfg`).

Configuration des données de test dans Lizmap

Aller sur le site <http://lizmap.local/>

Se connecter en tant qu'admin (rappel : mdp = `anf2023`)

Aller dans Administration > Lizmap configuration

Créer un nouveau repository (= un répertoire où sont enregistrés des projets QGIS configurés pour Lizmap) avec comme caractéristiques :

- Id = public
- Label = ANF SIST Géomatique 2023
- Local folder path = /home/sist/data/mon_travail/3_lizmap/public
- View project in this repo = anonymous, admins, users

Modify the repository

Id public

Label* ANF SIST Géomatique 2023

Local folder path* /var/www/lizmap-web-client/repos

☒ Allow themes/javascript codes for this repository

View projects in the repository ☒ anonymous
☒ admins (admins)
☒ users (users) [default group]

Display projects WMS links ☐ anonymous
☐ admins (admins)
☐ users (users) [default group]

Use the Edition tool ☐ anonymous
☐ admins (admins)
☐ users (users) [default group]

Allow export of layers ☐ anonymous
☐ admins (admins)
☐ users (users) [default group]

Always see complete layers data, even if filtered by login ☐ anonymous
☐ admins (admins)
☐ users (users) [default group]

Save

Création d'un lien symbolique pour le TP Leaflet

```
mkdir -p data/mon_travail/3_leaflet
cd /var/www/html
sudo ln -s /home/sist/data/mon_travail/3_leaflet 3_leaflet
```

Puis télécharger <https://github.com/dhg/Skeleton/releases/download/2.0.4/Skeleton-2.0.4.zip> et dézipper le fichier dans `/home/sist/public_html`. La page index.html est accessible par l'URL http://localhost/3_leaflet/.

