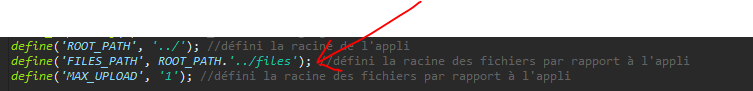
# Explorateur de fichiers

## 1- installation et configuration

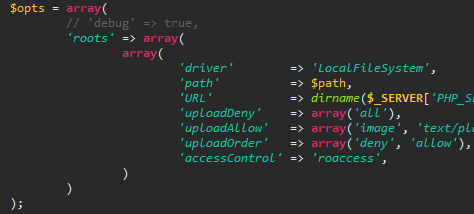
Récupérer le zip sur github : <https://github.com/pom35/echangesAqualeha>

L'extraire sur le serveur.

Créer le dossier racine de l'explorateur. Les admins auront accès à tous les fichiers qui y seront déposés. Le path de ce dossier est défini dans php/connector.php



Il est possible de paramétrer plusieurs dossiers racine. Pour cela, ajouter un tableau supplémentaire dans le tableau « roots ».



L’explorateur s’accède via l’url : echangesAqualeha/elfinder.php. On arrive sur la page de login.

## 2- Gestion des droits

La page de login va chercher les données sur le LDAP, dans login.php



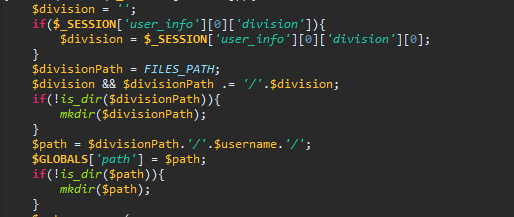
Un statut admin est paramétré dans connector.php, pour ajouter un admin il faut ajouter son login LDAP dans ce tableau :



Les admins ont accès à tous les fichiers depuis la racine.

Lorsqu’un user non admin se connecte, l’application vérifie la présence de son dossier et en fait la racine. Si le dossier n’existe pas, il est créé.

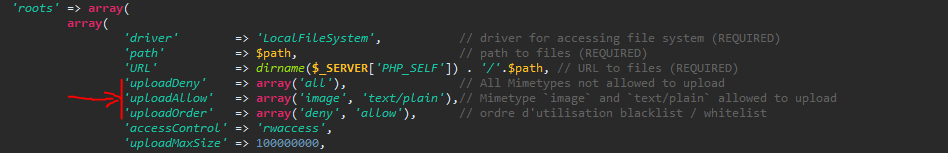
Le path est le suivant : racine\_des\_fichiers/{division/}/login.



## 3- ajout / suppression de types mime autorisés

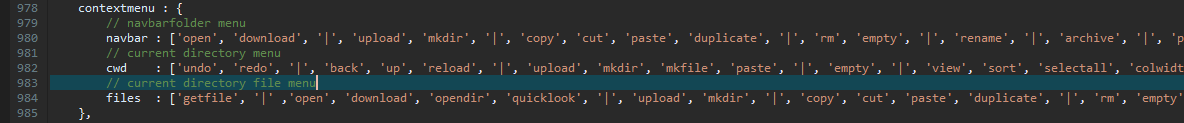
Pour paramétrer les types mime autorisés, il faut modifier les lignes uploadDeny et uploadAllow dans le tableau « roots ».

Dans l’exemple on interdit tous les types, puis on autorise les images et les fichiers texte.



## 4- ajouter / supprimer des options (bouttons)

La toolbar et le menu contextuel sont paramétrables dans elfinder.options.js



## 5- fonctionnement du login

### - par login / mdp ldap

Sur la page de login, un utilisateur peut se connecter avec son login/mdp LDAP, il aura accès à son dossier. S’il est configuré comme admin, il aura accès à tout le dépôt.

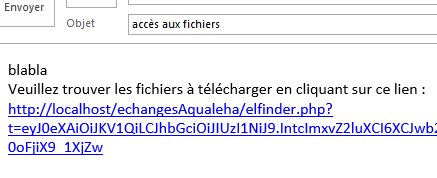
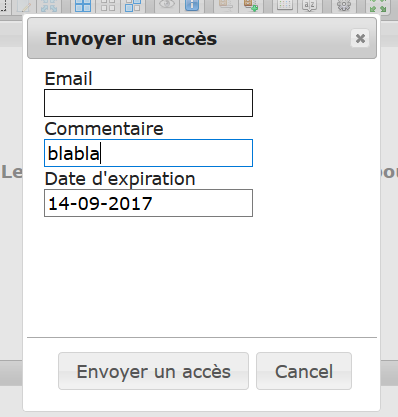
### - par token

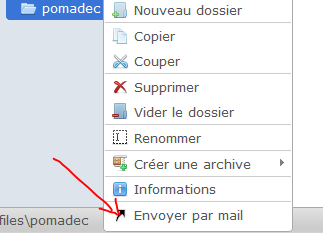
Il est possible de générer un token permettant d’accéder à un dossier. Pour cela, il faut cliquer droit sur un dossier et sélectionner envoyer par mail.

Une popup s’ouvre, demandant l’email, un commentaire et une date d’expiration. Seule la date d’expiration est obligatoire.

Le bouton « envoyer un accès » génère un lien mailto avec un token généré grâce à JWT, contenant le path vers le dossier partagé, le nom du dossier et la date d’expiration.

Ces infos sont testées sur la page login, et si elles sont valides (le dossier existe et le token n’est pas expiré), la personne pourra accéder au dossier en lecture seule.





## 6- commandes

### - drag and drop

L’explorateur supporte le drag and drop, il est possible d’uploader un fichier en le déposant dans le navigateur.

À l’intérieur de l’appli, un drag and drop simple permet de déplacer les fichiers. En appuyant sur la touche maj, ctrl ou alt, le fichier sera copié dans la destination. La touche alt permet également de sortir du navigateur pour télécharger le fichier.

## 7- logs

### - path

Par défaut, l’appli enregistre les logs des actions utilisateurs dans le dossier « .temp » à la racine des fichiers.

La valeur est affectée dans connector.php



### - actions

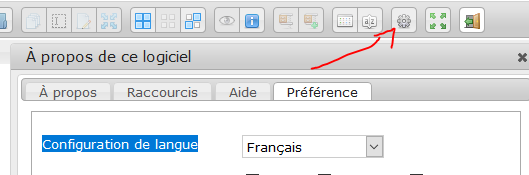
Les actions qui lancent des logs sont « bind » dans connector.php



« file » enregistre les téléchargements.

## 8- Préférences

Les préférences sont accessibles dans la barre de menu, on peut notamment y modifier la langue.



## 9- Génération du token

Le token de connexion est généré dans token.php grâce à Json Web Token (JWT).

Il contient le nom du dossier cible, le path vers le dossier et la date d’expiration et les crypte avec la clé définie.

Ce token est décrypté sur la page de login qui teste l’existence du dossier et l’expiration de la date.

