

Installation d'un service de stockage en ligne (Dropbox like)



Seafile est un logiciel libre permettant de construire un service de stockage.

Il a été cité dans les médias comme un système permettant de « mieux protéger sa vie privée sur Internet ».

La société qui développe le logiciel a développé une offre commerciale en mode SaaS sous le nom de Seacloud.

Seafile est constitué de trois composants :

- un serveur, Seafile (sous licence GPLv2)
- une interface web, SeaHub (sous licence Apache), permettant de consulter les fichiers gérés par Seafile directement via le web
- un client de synchronisation (disponible sur Android/Linux/MAC OS/Windows).

Le projet utilise un modèle inspiré de Git pour la gestion de fichiers, avec certaines adaptations, permettant par exemple de gérer de façon plus performante les gros fichiers.

Je vais décrire une installation de Seafile sur Debian.

Prérequis :

Pour commencer il faut installer certain package, pour cela j'ai choisi d'installer seafile avec mysql comme SGBD:

```
aptitude install python2.7 libpython2.7 python-setuptools python-imaging  
python-ldap python-urllib3 ffmpeg python-pip python-mysqldb python-memcache  
python-urllib3  
pip install pillow moviepy
```

Mais, j'ai installé cela dans le répertoire opt, j'ai créé un répertoire seafile dans le répertoire opt.

```
mkdir /opt/seafile  
cd /opt/seafile
```

Récupérez l'archive [ici](#).

Déplacez l'archive dans le répertoire /opt/seafile, puis décompressez l'archive.

```
tar -xvzf seafile-server*.tar.gz  
cd seafile-server*
```

Installation des services :

L'installation va commencer, il faut juste exécuter le script `setup-seafile-mysql.sh`.

Pendant l'installation des questions vous seront posés, vous devrez répondre aux questions (création de la base, nom, etc).

Je vous propose mon exemple qui suit:

```
# ./setup-seafiler-mysql.sh
Checking python on this machine ...
Checking python module: setuptools ... Done.
Checking python module: python-imaging ... Done.
Checking python module: python-mysqldb ... Done.

-----
This script will guide you to setup your seafiler server using MySQL.
Make sure you have read seafiler server manual at

    https://github.com/haiwen/seafiler/wiki

Press ENTER to continue
-----

What is the name of the server? It will be displayed on the client.
3 - 15 letters or digits
[ server name ] myfile

What is the ip or domain of the server?
For example: www.mycompany.com, 192.168.1.101
[ This server's ip or domain ] 172.10.4.102

Where do you want to put your seafiler data?
Please use a volume with enough free space
[ default "/opt/seafiler/seafiler-data" ]

Which port do you want to use for the seafiler filesaver?
[ default "8082" ]

-----
Please choose a way to initialize seafiler databases:
-----

[1] Create new ccnet/seafiler/seahub databases
[2] Use existing ccnet/seafiler/seahub databases

[ 1 or 2 ] 1

What is the host of mysql server?
[ default "localhost" ]

What is the port of mysql server?
[ default "3306" ]

What is the password of the mysql root user?
```

```
[ root password ]

verifying password of user root ... done

Enter the name for mysql user of seafile. It would be created if not exists.
[ default "root" ] seafile

Enter the password for mysql user "seafile":
[ password for seafile ]

Enter the database name for ccnet-server:
[ default "ccnet-db" ]

Enter the database name for seafile-server:
[ default "seafile-db" ]

Enter the database name for seahub:
[ default "seahub-db" ]

-----
This is your configuration
-----

server name:          myfile
server ip/domain:     172.10.4.102

seafile data dir:     /opt/seafile/seafile-data
fileserver port:      8082

database:             create new
ccnet database:        ccnet-db
seafile database:      seafile-db
seahub database:       seahub-db
database user:         seafile

-----
Press ENTER to continue, or Ctrl-C to abort
-----

Generating ccnet configuration ...

done
Successfully create configuration dir /opt/seafile/ccnet.
Generating seafile configuration ...

Done.
done
Generating seahub configuration ...
```

```
-----  
Now creating seahub database tables ...  
-----  
  
creating seafiler-server-latest symbolic link ... done  
  
-----  
Your seafiler server configuration has been finished successfully.  
-----  
  
run seafiler server:      ./seafiler.sh { start | stop | restart }  
run seahub server:      ./seahub.sh { start <port> | stop | restart <port>  
}  
  
-----  
If you are behind a firewall, remember to allow input/output of these tcp  
ports:  
-----  
  
port of seafiler fileserver:  8082  
port of seahub:               8000  
  
When problems occur, Refer to  
  
    https://github.com/haiwen/seafiler/wiki  
  
for information.
```

Lancement des services :

Ensuite, il faut lancer le service seafiler-server :

```
# ./seafiler.sh start  
  
[04/09/16 19:22:57] ../common/session.c(132): using config file  
/opt/seafiler/conf/ccnet.conf  
Starting seafiler server, please wait ...  
Seafiler server started  
  
Done.
```

Puis, le serveur web, comme c'est le premier lancement, vous allez créer votre login d'administration, répondez correctement au questionnaire, mais aussi choisir le port du serveur web (voici, mon exemple) :

```
# ./seahub.sh start 8000
```

```
Starting seahub at port 8000 ...
```

```
-----  
It's the first time you start the seafile server. Now let's create the admin  
account  
-----
```

```
What is the email for the admin account?  
[ admin email ] root@mydomain.com
```

```
What is the password for the admin account?  
[ admin password ]
```

```
Enter the password again:  
[ admin password again ]
```

```
-----  
Successfully created seafile admin  
-----
```

```
Seahub is started
```

```
Done.
```

Voici, votre serveur Seafile est prêt, vous pouvez l'administrer au travers de l'interface web dans mon exemple <http://172.10.4.102:8000>

Pour arrêter le service seafile :

```
./seahub.sh stop  
./seafile.sh stop
```

puis, le relancer :

```
./seafile.sh start  
./seahub.sh start
```

Automatisation :

Si, vous voulez automatiser le lancement des services au démarrage, je vais vous montrer comment faire.

Création du login seafile :

```
groupadd seafile  
useradd -g seafile -s /bin/sh -c "seafile" seafile
```

Changer le propriétaire des fichiers seafile :

```
chown -R seafile:seafile /opt/seafile/*
```

puis, faire le script de lancement suivant (avec les privilèges root) :

```
# vim /etc/init.d/seafiler
```

Voici, le contenu du script

```
#!/bin/sh

### BEGIN INIT INFO
# Provides:          seafiler
# Required-Start:    $remote_fs $syslog
# Required-Stop:     $remote_fs $syslog
# Default-Start:     2 3 4 5
# Default-Stop:
# Short-Description: seafiler
### END INIT INFO

SEAFILER_REP=/opt/seafiler/seafiler-server-latest
SEAFILER_BIN=${SEAFILER_REP}/seafiler.sh
SEAHUB_BIN=${SEAFILER_REP}/seahub.sh
USER=seafiler
. /lib/lsb/init-functions

if [ ! -x $SEAFILER_BIN ]; then
    log_daemon_msg "${SEAFILER_BIN} does not exist."
    err=1
    case "$err" in
        0) log_end_msg 0;;
        *) log_end_msg 1; exit 1;;
    esac
fi
if [ ! -x $SEAHUB_BIN ]; then
    log_daemon_msg "${SEAHUB_BIN} does not exist."
    err=1
    case "$err" in
        0) log_end_msg 0;;
        *) log_end_msg 1; exit 1;;
    esac
fi

start() {
    log_daemon_msg "lancement de $SEAFILER_BIN"
    su $USER -c "$SEAFILER_BIN start"
    case "$?" in
        0) log_end_msg 0;;
        *) log_end_msg 1; exit 1;;
    esac
    log_daemon_msg "lancement de $SEAHUB_BIN"
    su $USER -c "$SEAHUB_BIN start"
```

```
        case "$?" in
            0) log_end_msg 0;;
            *) log_end_msg 1; exit 1;;
        esac
    }
    stop() {
        log_daemon_msg "lancement de $SEAFHUB_BIN"
        su $USER -c "$SEAFHUB_BIN stop"
        case "$?" in
            0) log_end_msg 0;;
            *) log_end_msg 1; exit 1;;
        esac
        log_daemon_msg "lancement de $SEAFHUB_BIN"
        su $USER -c "$SEAFHUB_BIN stop"
        case "$?" in
            0) log_end_msg 0;;
            *) log_end_msg 1; exit 1;;
        esac
    }
    case "$1" in
        start)
            start
            ;;
        stop)
            stop
            ;;
        restart)
            stop
            start
            ;;
        *)
            echo $"Usage: seafhub {start|stop|restart}"
            ;;
    esac
```

Attribuer les droit suivants :

```
chmod 755 /etc/init.d/seafhub
```

Lancement des services au démarrage de linux, saisir la commande suivante :

```
update-rc.d seafhub defaults
```

From:
<https://www.ksh-linux.info/> - **Know Sharing**

Permanent link:
<https://www.ksh-linux.info/stockage/seafhub/installation-d-un-service-de-stockage-en-ligne>

Last update: **14/02/2018 10:05**

