

Open clinica sur Raspberry pi

Installation

Caractéristiques de la machine

- 768 Mo de RAM minimum (1 Go ou plus recommandé)
- 4 Go de disque minimum (16 Go recommandés)

Installez le serveur

Installez :

- le serveur Tomcat
- le serveur PostgreSQL

Installation des paquetages : *

```
sudo apt-get update
sudo apt-get dist-upgrade
sudo apt-get install postgresql tomcatv tomcatv-admin pwgen libpg-java
openjdk-v-jdk
```



remplacez **v** par le numéro de version (s'aider de l'auto-complétion avec Tab ↴)

Configurez Tomcat

*

```
<listener><listener-class>php.java.servlet.ContextLoaderListener</listener-
class></listener>
<servlet><servlet-name>PhpJavaServlet</servlet-name><servlet-
class>php.java.servlet.PhpJavaServlet</servlet-class>
</servlet>
<servlet><servlet-name>PhpCGIServlet</servlet-name><servlet-
class>php.java.servlet.fastcgi.FastCGIServlet</servlet-class>
<init-param><param-name>prefer_system_php_exec</param-name><param-
value>On</param-value></init-param>
<init-param><param-name>php_include_java</param-name><param-
value>Off</param-value></init-param>
```

```
</servlet>
<servlet-mapping><servlet-name>PhpJavaServlet</servlet-name><url-
pattern>*.phpjavabridge</url-pattern> </servlet-mapping>
<servlet-mapping><servlet-name>PhpCGIServlet</servlet-name><url-
pattern>*.php</url-pattern></servlet-mapping>
```

Installation

Installez java, postgresql (aidez-vous de l'auto-complétion par la touche **Tab ↹** pour les versions) :

- sudo apt-get update
- sudo apt-get upgrade
- sudo apt-get install openjdk-vx-jdk postgresql phppgadmin -y

Téléchargement

Téléchargez Tomcat sur <http://tomcat.apache.org/download-70.cgi>

The screenshot shows the Apache Tomcat 7.0.52 download page. At the top, it says "7.0.52". Below that, a message reads: "Please see the [README](#) file for packaging information. It e...". Under the heading "Binary Distributions", there is a list:

- Core:
 - [zip \(pgp, md5\)](#)
 - [tar.gz \(pgp, md5\)](#)
 - [32-bit Windows zip \(pgp, md5\)](#)
 - [64 bit Windows zip \(pgp, md5\)](#)

Dans **Binary Distributions/Core**, choisir le lien **tar.gz**.

Le placer, par exemple, dans le dossier **/home/\$USER/Téléchargements**.

Pour l'installer sur un Raspberry, le copier via scp vers un répertoire du Raspberry Pi :



- `scp ~/Téléchargements/apache-tomcat-[version].tar.gz pi@framboise:/home/pi`

Téléchargez le logiciel de OpenClinica sur le site :

<https://community.openclinica.com/project/openclinica>.



Pour télécharger le logiciel, vous devez avoir un compte utilisateur de la communauté de OpenClinica, à créer gratuitement. Vous devrez vous connecter au compte pour télécharger le logiciel de OpenClinica

Dé-comprimez le fichier **OpenClinica-v.x.y.z.zip** téléchargé → un répertoire **OpenClinica-v.x.y.z** apparaît.

Installez java

Créez le lien symbolique : *

```
sudo ln -s /usr/lib/jvm/jdk* /usr/local/java
```

Vous pouvez vérifier par : *

```
ll /usr/local
```

Le lien doit apparaître.

Installez Tomcat

Allez dans le répertoire de téléchargement et décompressez l'archive téléchargée :

- cd /home/\$USER/Téléchargements
sudo tar -zxvf apache-tomcat-v.x.y.tar.gz

Un répertoire **apache-tomcat-v.x.y/** apparaît.

Déplacez ce répertoire vers **/usr/local/** et créez le lien **tomcat** :

- sudo mv apache-tomcat-v.x.y/ /usr/local/
sudo ln -s /usr/local/apache-tomcat-v.x.y /usr/local/tomcat

Créez l'utilisateur **tomcat** avec le mot de passe **tomcat** :

- sudo adduser tomcat

```
Ajout de l'utilisateur « tomcat » ...
Ajout du nouveau groupe « tomcat » (1004) ...
Ajout du nouvel utilisateur « tomcat » (1001) avec le groupe « tomcat » ...
Le répertoire personnel « /home/tomcat » existe déjà. Rien n'est copié depuis «
/etc/skel ».
Entrez le nouveau mot de passe UNIX :
Retapez le nouveau mot de passe UNIX :
passwd : le mot de passe a été mis à jour avec succès
Modification des informations relatives à l'utilisateur tomcat
Entrez la nouvelle valeur ou « Entrée » pour conserver la valeur proposée
    Nom complet []:
    N° de bureau []:
    Téléphone professionnel []:
    Téléphone personnel []:
    Autre []:
Cette information est-elle correcte ? [0/n]
```

Et rendez-le propriétaire des répertoires de tomcat :

- cd /usr/local/tomcat
sudo chown -R tomcat:tomcat .
sudo chmod -R 6770 .

Pour pouvoir intervenir sans être root, rendre l'utilisateur **\$USER** membre du groupe **tomcat** :

- sudo usermod -aG tomcat \$USER



Attention à mettre le **-a** : sinon, \$USER changera de groupe
au lieu d'ajouter tomcat

Pour définir le mot de passe de l'utilisateur **tomcat**, ouvrez le fichier
/usr/local/tomcat/conf/tomcat-users.xml en ajoutant ces lignes (entre les balises **tomcat-users**):

</usr/local/tomcat/conf/tomcat-users.xml>

```
...
<tomcat-users>
    <role rolename="tomcat"/>
    <role rolename="admin-gui"/>
    <role rolename="manager-gui"/>
    <user username="tomcat" password="tomcat"
roles="tomcat,admin-gui,manager-gui"/>
</tomcat-users>
...
```

L'utilisateur est **tomcat**, son mot de passe est **tomcat**, ses rôles sont **tomcat, admin-gui** et

manager-gui.

Démarrez le serveur Tomcat:

- `sudo /usr/local/tomcat/bin/startup.sh`

Ouvrez dans un navigateur l'adresse

- en local : <http://localhost:8080>
- sur internet : http://<adresse_du_site>:8080

La page de tomcat doit s'afficher :

Pour que tomcat se lance automatiquement à chaque démarrage, recopiez le fichier
`~/Téléchargements/OpenClinica-v.x.y.z/install-docs/linux/tomcat` :

- `cd /etc/init.d/
sudo cp ~/Téléchargements/OpenClinica-v.x.y.z/install-docs/linux/tomcat .`

Rendez-le exécutable et ajoutez-le à la liste des programmes au démarrage :

- `sudo chmod a+x /etc/init.d/tomcat
sudo update-rc.d tomcat defaults`

Installez postgresql et phppgadmin

Un utilisateur système postgres a été créé à l'installation. Il n'a pas de mot de passe : c'est un utilisateur bloqué et le mieux est qu'il le reste.

utilisateur postgres Au départ, toutes les opérations d'administration se font donc avec l'utilisateur **postgres** qui seul peut se connecter.

Pour vous connecter en tant qu'utilisateur **postgres**, utilisez la commande sudo :

- `sudo -i -u postgres`

L'invite de commande mentionne que vous êtes actif en tant que postgres :

```
pi@framboise ~ $ sudo -i -u postgres  
postgres@framboise:~$
```



A la fin de cette session d'administration dans PostgreSQL, il suffira de taper

- `exit`

pour reprendre la main en tant qu'utilisateur du système.

On peut aussi lancer une commande (ici **psql**) en tant que postgres :

- `sudo -u postgres psql`

Dès que cette commande se termine, on se retrouve avec l'utilisateur habituel.

Création d'un nouvel utilisateur pour gérer postgresql

Nous allons créer l'utilisateur qui nous permettra de gérer postgresql.

Pour cela, lancez l'outil d'administration de PostgreSQL en tant qu'utilisateur **postgres** :

- `sudo -u postgres psql`

```
pi@framboise ~ $ sudo -i -u postgres
postgres@framboise:~$ psql
psql (9.1.12)
Saisissez « help » pour l'aide.

postgres=#
```

Vous êtes dans une interface en ligne de commande, en tant que **root** (# en bout de ligne).

Pour créer un environnement plus sûr, créez un rôle pour l'administration, avec toutes les autorisations sauf l'héritage des droits :

- `CREATE ROLE <admin> LOGIN ENCRYPTED PASSWORD '<mot_de_passe>' SUPERUSER NOINHERIT CREATEDB CREATEROLE;`

<admin>
nom d'utilisateur
<mot_de_passe>
mot de passe

N'oubliez pas le point-virgule en fin de ligne !

Quittez l'environnement psql :

- `\q`

Vérifiez en essayant de vous connecter avec l'utilisateur que vous venez de créer :

- `sudo -i -u postgres`

Installation de l'interface web PhpPgAdmin

Installons maintenant l'interface web **PhpPgAdmin** pour PostgreSQL qui simplifie grandement la mise en place et l'administration.

Ouvrez avec les droits d'administration le fichier `/etc/lighttpd/lighttpd.conf` et ajoutez la ligne suivante à la fin du fichier :

</etc/lighttpd/lighttpd.conf>

```
...
alias.url += ( "/phppgadmin" => "/usr/share/phppgadmin/" )
```

L'adresse <http://monsite.tld/phppgadmin> pointe maintenant sur le bon répertoire interne.

Dans le fichier **/etc/lighttpd/lighttpd.conf**, assurez-vous que le module **mod_alias** est décommenté (donc activé)

et activez les modules **mod_fastcgi** et **mod_cgi** :

- sudo lighty-enable-mod fastcgi cgi

Redémarrez le serveur :

- sudo service lighttpd restart

Allez à la page de phpPgAdmin : <http://monsite.tld/phppgadmin>

Cliquez sur **PostgreSQL** sur le côté gauche.

Dans la fenêtre de connexion, entrez le nom et le mot de passe de l'utilisateur créé plus haut pour gérer postgresql. Cela devrait vous connecter

The screenshot shows the phpPgAdmin interface. On the left, there's a sidebar titled "phpPgAdmin" with a tree view of servers and databases. Under "PostgreSQL", it shows a "postgres" database with "Schémas" and "Slony" sub-folders. The main panel displays a message: "PostgreSQL 9.1.12 lancé sur localhost:5432 -- Vous êtes connecté avec le profil « paul »". Below this, there's a table for managing databases, with one row for "postgres". The table has columns: Base de données, Propriétaire, Codage, Tri, Type de caractère, Tablespace, Taille, Actions, and Rôles. The "Actions" column contains links for "Supprimer", "Droits", and "Modifier". At the bottom of the main panel, there are buttons for "Sélectionner tout / Désélectionner tout", "Lancer", and a link to "Créer une base de données".

Pour ajouter un rôle (utilisateur)

Cliquez sur **rôles** dans la zone du milieu en haut de la page

Cliquez sur **Créer Rôle**

Créer nom d'utilisateur / mot de passe et donner toutes les autorisations autres que **privileges Inherits**.

Ignorer les autres options dans les cases à cocher

Cliquez sur **Créer**

From:

<https://nfrappe.fr/doc-0/> - Documentation du Dr Nicolas Frappé



Permanent link:

<https://nfrappe.fr/doc-0/doku.php?id=logiciel:internet:openclinica:ocfnf>

Last update: **2022/08/13 21:57**