

# Installer Gogs sur Debian jessie à partir du paquet avec Mysql et Apache

- Objet : Gogs est un service Web de gestion de dépôts Git, comme Github ou Gitlab
- Niveau requis :  
[avisé](#)
- Commentaires : Installation et configuration à partir du paquet des dépôts Packager.io
- Débutant, à savoir : Pas mal de trucs quand même 😊
- Tags :  
[en-chantier](#)

## Introduction

Il existe plusieurs façon d'installer Gogs. Depuis les sources, depuis une archive, ou depuis un dépôt externe hébergé sur Packager.io. Ce tuto traitera de ce dernier cas.

Gogs nécessite l'utilisation d'une base de donnée. Sqlite3 (sauf pour l'installation via le paquet), Postgree et Mysql sont utilisable. Nous ne traiterons ici que de Mysql

Gogs utilise son propre serveur web écoutant sur un port dédié. Nous utiliserons ici Apache2.4 comme proxy pour rediriger les connexions vers ce port.

Dans le suite du tuto, on va considérer que le serveur est accessible à l'adresse 12.34.56.78 via l'url <http://chez-moi.net>.

## Dépendances

Apache2, git > 1.7 et mysql-server doivent êtres installés.

## Installation

### Le paquet Gogs

```
wget -q0 - https://deb.packager.io/key | apt-key add -  
echo "deb https://deb.packager.io/gh/pkgr/gogs jessie pkgr" >  
/etc/apt/sources.list.d/gogs.list  
sudo apt-get update  
sudo apt-get install gogs
```

Le paquet créer automatiquement l'utilisateur gogs, et les dépôts git seront créés par défaut dans /home/gogs/gogs-repositories. L'application est installée dans /opt/gogs. Les fichiers de configuration se trouvent dans /etc/gogs/. Le paquet créé aussi une série de fichier dans /etc/systemd/system/ permettant de gérer le lancement et l'arrêt de l'application via systemd.

## Création de la base de donnée

```
mysql -uroot -p
```

```
CREATE DATABASE gogs;  
GRANT ALL privileges ON gogsuser.* TO gogs@'%' IDENTIFIED BY  
'votreSuperMotdePasse';  
FLUSH privileges;  
quit;
```

## Configuration de Apache2

### Les modules à activer

mod\_proxy\_html, mod\_proxy\_http, et mod\_proxy\_connect pour le https

```
a2enmod proxy proxy_connect
```

### Le fichier Vhost

Par défaut, l'application est accessible sur le port 3000. Par exemple <http://chez-moi.net:3000>. Mais, on souhaite y accéder à l'adresse <http://chez-moi.net/git>.

Pour cela, on va utiliser apache comme *mandataire/passerelle*. Dans votre fichier de configuration de Apache2, ajouter :

#### pour http

```
SSLProxyEngine on  
<Location /git>  
    Require all granted  
    ProxyPass http://12.34.56.78:3000  
    ProxyPassReverse http://12.34.56.78:3000  
</Location>
```

#### pour https

```
SSLProxyEngine on  
SSLProxyVerify none  
SSLProxyCheckPeerCN off  
SSLProxyCheckPeerName off  
  
<Location /git>
```

```
Require all granted
ProxyPass http://12.34.56.78:3000
ProxyPassReverse http://12.34.56.78:3000
</Location>
```

## Le fichier de configuration de gogs

Il est situé dans `/etc/gogs/conf/app.ini`. Les commentaires sont assez explicites.

Il y a juste un petit piège à éviter pour le `https` avec `ReverseProxy`, il ne faut pas spécifier le protocole `https`, ni renseigner les chemin des certificats. Je met donc juste la section `server` avec le petit morceau de config, où il s'agit juste d'indiquer le protocole `https` dans `ROOT_URL` 😊

nom

```
[server]
PROTOCOL = http
DOMAIN = chez-moi.net
; ROOT_URL = %(PROTOCOL)s://%(DOMAIN)s:%(HTTP_PORT)s/
; Config perso pour le https avec reverseProxy
ROOT_URL = https://%(DOMAIN)s:%(HTTP_PORT)s/
HTTP_ADDR = 0.0.0.0
HTTP_PORT = 3000
; Permission for unix socket
UNIX_SOCKET_PERMISSION = 666
; Local (DMZ) URL for Gogs workers (such as SSH update) accessing web
service.
; In most cases you do not need to change the default value.
; Alter it only if your SSH server node is not the same as HTTP node.
LOCAL_ROOT_URL = %(PROTOCOL)s://%(HTTP_ADDR)s:%(HTTP_PORT)s/
; Disable SSH feature when not available
DISABLE_SSH = false
; Whether use builtin SSH server or not.
START_SSH_SERVER = false
; Domain name to be exposed in clone URL
SSH_DOMAIN = %(DOMAIN)s
; Port number to be exposed in clone URL
SSH_PORT = 22
; Port number builtin SSH server listens on
SSH_LISTEN_PORT = %(SSH_PORT)s
; Root path of SSH directory, default is '~/ssh', but you have to use
'/home/git/.ssh'.
SSH_ROOT_PATH =
; Directory to create temporary files when test public key using ssh-
keygen,
; default is system temporary directory.
SSH_KEY_TEST_PATH =
; Path to ssh-keygen, default is 'ssh-keygen' and let shell find out
which one to call.
SSH_KEYGEN_PATH = ssh-keygen
```

```
; Indicate whether to check minimum key size with corresponding type
MINIMUM_KEY_SIZE_CHECK = false
; Disable CDN even in "prod" mode
OFFLINE_MODE = false
DISABLE_ROUTER_LOG = false
; Generate steps:
; $ ./gogs cert -ca=true -duration=8760h0m0s -host=myhost.example.com
;
; Or from a .pfx file exported from the Windows certificate store (do
; not forget to export the private key):
; $ openssl pkcs12 -in cert.pfx -out cert.pem -nokeys
; $ openssl pkcs12 -in cert.pfx -out key.pem -nocerts -nodes
CERT_FILE = custom/https/cert.pem
KEY_FILE = custom/https/key.pem
; Upper level of template and static file path
; default is the path where Gogs is executed
STATIC_ROOT_PATH =
; Default path for App data
APP_DATA_PATH = data
; Application level GZIP support
ENABLE_GZIP = false
; Landing page for non-logged users, can be "home" or "explore"
LANDING_PAGE = home
```

## Utilisation

From:  
<http://debian-facile.org/> - **Documentation** - Wiki

Permanent link:  
<http://debian-facile.org/utilisateurs:bendia:tutos:installer-gogs-sur-debian-jessie-avec-apache-et-mysql>



Last update: **02/11/2016 12:28**