



# Autogestión: Montar y Administrar gitolite + cgkit + git-daemon

👤 DRYMER

📅 JULIO 5

💬 9 COMENTARIOS

📁 SOFTWARE

Los repositorios git son una de las mejores cosas que se han hecho a nivel de compartir. No sólo permite descargar y hojear el código sino que además permite ver que cambios se han hecho cuando, aumentando así la confianza que se pueda tener en ese programa. Así que al lío. **Veremos cómo se instala en un servidor con Debian Wheezy.**

La parte de la instalación es fácil:

```
# aptitude install git gitolite3
```

Nos pedirá una clave SSH. Podemos pasar por que vamos a re-configurar el paquete, para ajustar algunas cosillas.

```
# dpkg-reconfigure gitolite3
```

Nos preguntará que usuario vamos a usar, directorio home de ese usuario (dónde estarán los repositorios) y una clave SSH. Deberíamos poner la de nuestro ordenador. Una vez hecho,

entraremos con el usuario git y iniciaremos el repositorio.

```
# su git
$ gitolite setup -a admin
```

Veremos que en el home de git aparecen ya dos repositorios git, uno llamado testing y el otro gitolite-admin. Este es muy importante, por eso lo clonaremos a nuestro ordenador y administraremos todos los repositorios desde ahí.

```
$ git clone git@servidor.git.lol:gitolite-admin
$ cd gitolite-admin
```

Aquí veremos dos directorios, keys y conf. En el primero veremos un archivo llamado admin.pub, que contiene nuestra clave de administrador. En conf veremos gitolite.conf, que es dónde está la magia. Ahí añadiremos los repositorios y permisos de estos que queramos.

Ahora, según las necesidades de cada persona, habrán distintas necesidades. Para una mejor comprensión, mejor leer la [documentación](#). A continuación habrá un pequeño resumen con una pequeña explicación a modo de comentario.

```
# gitolite.conf

# Declaramos los grupos que queramos
@usuarios = foo bar

# Necesario para que git-daemon acceda a los repositorios
repo @all
    R          = daemon

repo gitolite-admin
    RW+        = admin

# Declaramos repositorios. El que tengan el mismo prefijo (ElBinario), han
# secciones, cómo veremos más adelante con cgkit.
repo ElBinario/gnusrss
    RW+        = @elbinario
    R          = @all

repo ElBinario/eve
    RW+        = @usuarios
    R          = @all

repo ElBinario/alex
    RW+        = @usuarios
    R          = @all
```

```

repo ElBinario/bofh
    RW+      =    @usuarios
    R        =    @all

repo ElBinario/dni
    RW+      =    @usuarios
    R        =    @all

repo ElBinario/etherpad-lite-config-files
    RW+      =    @usuarios
    R        =    @all

repo ElBinario/asocial
    RW+      =    @usuarios
    R        =    @all

# Según lo que se quiera, esto puede no interesar. Esto permite a cualquier
# repositorio.
repo CREATOR/[a-z].*
    C        =    @usuarios
    RW+      =    CREATOR
    RW       =    WRITERS
    R        =    READERS

```

Tras modificar el fichero conf/gitolite.conf, tendremos que hacer constar los cambios:

```

$ git add conf/gitolite.conf
$ git commit -m "initial commit"
$ git push --set-upstream origin master # esto sólo la primera vez, las siguientes
$ git push

```

Ahora podríamos clonar de tal manera:

```

# Para clonar uno existente
$ git clone git@git.server.lol:ElBinario/eve.git
# Para crear un repositorio
$ git clone git@git.server.lol:drymer/testeando.git

```

Vale. Ahora le pondremos una interfaz web, que siempre queda más bonito. gitweb puede servir, pero para mi es demasiado sencillo, aunque tiene la ventaja de que anda en la

paquetería de debian, a diferencia de cgkit. Pero bueno, cgkit ofrece más cosas que gitweb, así que usaremos ese:

```
$ git clone git://git.zx2c4.com/cgkit
$ cd cgkit
$ git submodule init
$ git submodule update
$ make get-git
$ make
$ make install
```

A continuación la configuración. Primero el virtual host de Nginx, un script de manejo del cgi y luego el /etc/cgitrc, que es su fichero de configuración. Está todo en el [pad](#), para no alargar demasiado el artículo. No explicaré que hace cada línea del /etc/cgitrc por que sería muy largo. Para eso, mejor leer el [man](#).

Hecho esto, reiniciamos nginx e iniciamos el script de inicio (muy cutre) de fcgiwrap y ya deberíamos ver nuestro repositorio git. Al final he puesto un pequeño cheatsheet de cómo administrar diferentes aspectos necesarios.

 [AUTOGESTIÓN](#) [GIT](#) [SERVIDOR](#)

Compartir    

[« Previous Post](#)

[Los Mangarrianes: 3.- Unión Europea](#)

[Next Post »](#)

[Y otro servicio más: repositorios git](#)

Proudly powered by [WordPress](#) | Theme: [Theia Lite](#) by Gecodigital.