

TRABAJO PRÁCTICO FINAL

Laboratorio de redes y sistemas operativos

GITEA (<https://gitea.io>)

Your children are not your children. They are the sons and daughters of Life's longing for itself. They come through you but not from you, and though they are with you yet they belong not to you.

Kahlil Gibran

Gitea es un servicio de auto hosteo de Git. Es similar a GitHub, Bitbucket o Gitlab.

La meta de este proyecto es hacerlo más fácil, rápido, y más básica la forma de levantar un servicio de Git auto hosteado. Soporta todas las plataformas, incluyendo Linux, macOS y Windows, incluso en arquitecturas como ARM o PowerPC.

INSTALACIÓN

Hay varias formas de instalarlo¹, para distintos sistemas operativos. Elegimos instalarlo en el sistema operativo que tenemos corriendo en nuestra notebook, un Ubuntu 16.xx. A continuación agregamos una imagen con los detalles de la computadora.



Para instalar en Ubuntu (o cualquier Linux) no tenemos paquetes disponibles, así que debemos hacer la instalación mediante un archivo .BIN. Podemos elegir entre las distintas versiones cambiando la url donde va a buscar el archivo. Para saber

¹ Cabe destacar que en la página hay un documento explicando el proceso de instalación para cada uno de los sistemas operativos.

cual podemos elegir entramos a <https://dl.gitea.io/gitea/> y hacemos click en la carpeta que contiene la versión deseada² y copiamos la url para usarla a en el siguiente paso.

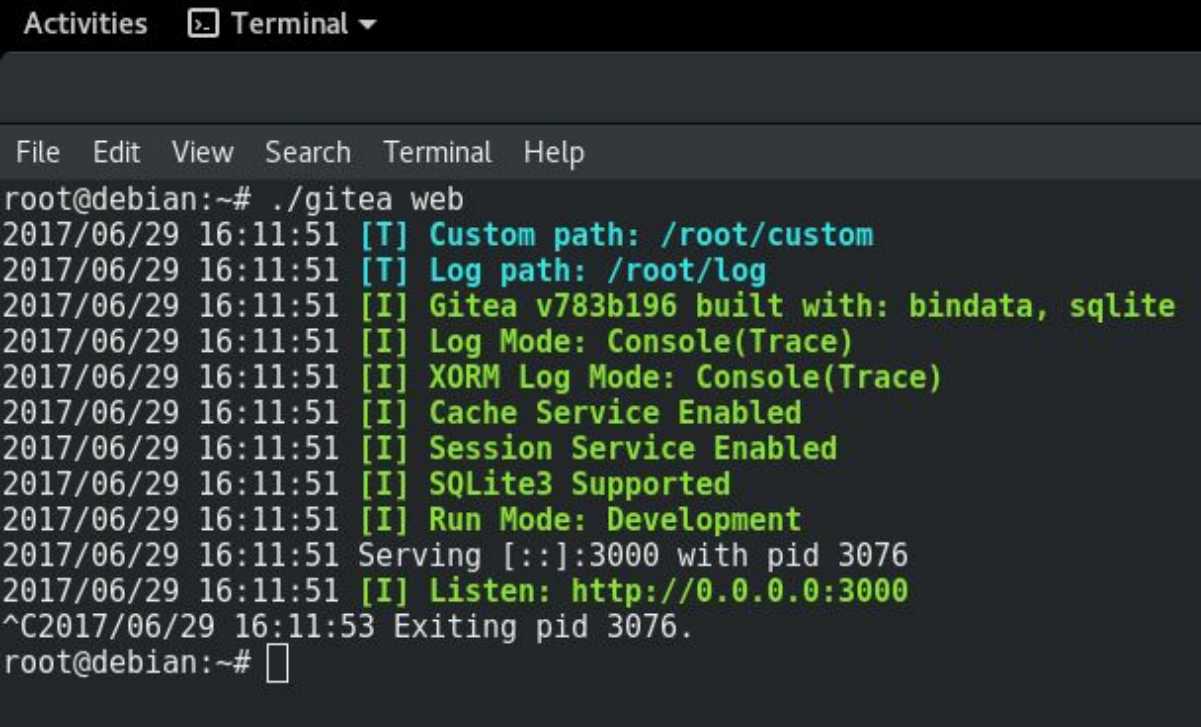
Una vez definida la versión abrimos una terminal con permisos de root y escribimos:

```
wget -O gitea https://dl.gitea.io/gitea/1.0.1/gitea-1.0.1-linux-amd64
chmod +x gitea
```

3

Modificando la url por la que copiamos en el paso previo.

Una vez hecho esto testeamos que no tengamos conflicto y llamamos al programa desde consola escribiendo `./gitea web`, si nos aparece en pantalla lo mismo que en la imagen que sigue vamos por el buen camino.



```
Activities [x] Terminal v
File Edit View Search Terminal Help
root@debian:~# ./gitea web
2017/06/29 16:11:51 [T] Custom path: /root/custom
2017/06/29 16:11:51 [T] Log path: /root/log
2017/06/29 16:11:51 [I] Gitea v783b196 built with: bindata, sqlite
2017/06/29 16:11:51 [I] Log Mode: Console(Trace)
2017/06/29 16:11:51 [I] XORM Log Mode: Console(Trace)
2017/06/29 16:11:51 [I] Cache Service Enabled
2017/06/29 16:11:51 [I] Session Service Enabled
2017/06/29 16:11:51 [I] SQLite3 Supported
2017/06/29 16:11:51 [I] Run Mode: Development
2017/06/29 16:11:51 Serving [::]:3000 with pid 3076
2017/06/29 16:11:51 [I] Listen: http://0.0.0.0:3000
^C2017/06/29 16:11:53 Exiting pid 3076.
root@debian:~# [ ]
```

² Para este trabajo elegimos la versión más actualizada, en la rama MASTER

³ “chmod +x” habilita todos los permisos, tanto a Usuario, Grupo y Otros

Problemas al levantar el servicio

Versiones antiguas de glibc

Distribuciones antiguas de Linux (como Debian 7 y CentOS 6) no podrán leer los binarios de Gitea, resultando en un error como `./gitea: /lib/x86_64-linux-gnu/libc.so.6: version `GLIBC_2.14' not found (required by ./gitea)`. Esto es debido al soporte de SQLite integrado en los binarios provistos. En el futuro, se proveerán binarios sin el requerimiento para glibc. Como workaround, podés actualizar tu distribución o [instalar desde el fuente](#).

Levantando gitea en otro puerto

Si estás teniendo un error como `702 runWeb() [E] Failed to start server: listen tcp 0.0.0.0:3000: bind: address already in use` gitea debería ser iniciado con otro puerto, que esté libre. Esto es posible usando: `./gitea web -p $PORT`.

Configuración

Una vez levantado el servicio, pasamos a la configuración del mismo. Para esto abrimos un explorador y escribimos <http://localhost:3000/> nos aparecerá una ventana como la siguiente:

The screenshot shows the 'Initial configuration' page of Gitea. It is divided into two main sections: 'Database Settings' and 'General Application Settings'.
Database Settings: Includes a dropdown for 'Database Type' (MySQL), a 'Host' field (127.0.0.1:3306), 'User' (gitea), 'Password' (empty), and 'Database Name' (gitea). A note mentions using the InnoDB engine with utf8_general_ci charset for MySQL.
General Application Settings: Includes 'Application Name' (Gitea: Git with a cup of tea), 'Repository Root Path' (/root/gitea-repositories), 'LFS Root Path' (empty), 'Run User' (root), 'Domain' (localhost), 'SSH Port' (22), 'HTTP Port' (3000), and 'Application URL' (http://localhost:3000/). Each field has a small explanatory text below it.

En estos campos hay que cambiar los valores de los campos que creamos convenientes, en principio se pueden dejar varios de los que vienen preestablecidos.

En nuestro caso puntual en **Database Settings** hicimos la siguiente configuración

Database Settings

Gitea requires MySQL, PostgreSQL, SQLite3, or TiDB.

Database Type *

Host *

User *

Password *

Database Name *

Todos los campos son obligatorios, el host dejamos el que venia por default. El campo “Database Name” requiere una base de datos existente con ese nombre.

De no existir y querer seguir la instalación con la configuración que hayamos hecho, nos aparecerá un error como el siguiente:

Initial configuration

Database setting is invalid: Error 1698: Access denied for user 'gitea'@'localhost'

If you are running Gitea inside Docker, please read the [guidelines](#) carefully before changing anything on this page.

Database Settings

Gitea requires MySQL, PostgreSQL, SQLite3, or TiDB.

Database Type *

Host *

User *

Password *

Database Name *

Please use the INNODB engine with utf8_general_ci charset for MySQL.

Para solucionar este problema y poder efectivizar la instalación, procedemos a crear la base de datos.

Abrimos una terminal y escribimos: `mysql -u usuario -p pass`.

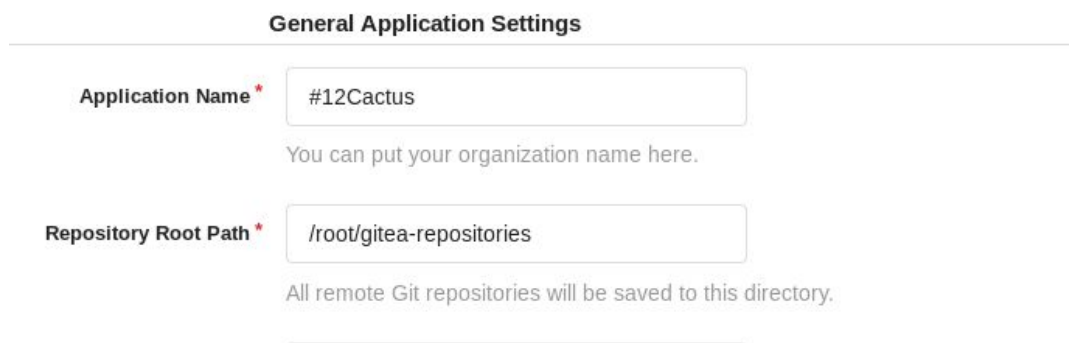
Debería cambiar el prompt por: "mysql>"

Ejecutamos el siguiente comando: `CREATE DATABASE dbName;`

Donde `dbName` en nuestro caso es "gitea".

Los campos User y Password deben coincidir con los de mysql.

En **General application Settings**, pusimos el nombre del grupo en *Application Name*, el siguiente campo indica donde ubicará la carpeta de los repositorios.



The screenshot shows the 'General Application Settings' page. It has two main input fields:

- Application Name ***: The input field contains the text "#12Cactus". Below the field is a greyed-out note: "You can put your organization name here."
- Repository Root Path ***: The input field contains the text "/root/gitea-repositories". Below the field is a greyed-out note: "All remote Git repositories will be saved to this directory."

El resto de las opciones fueron dejadas como vinieron preestablecidas.

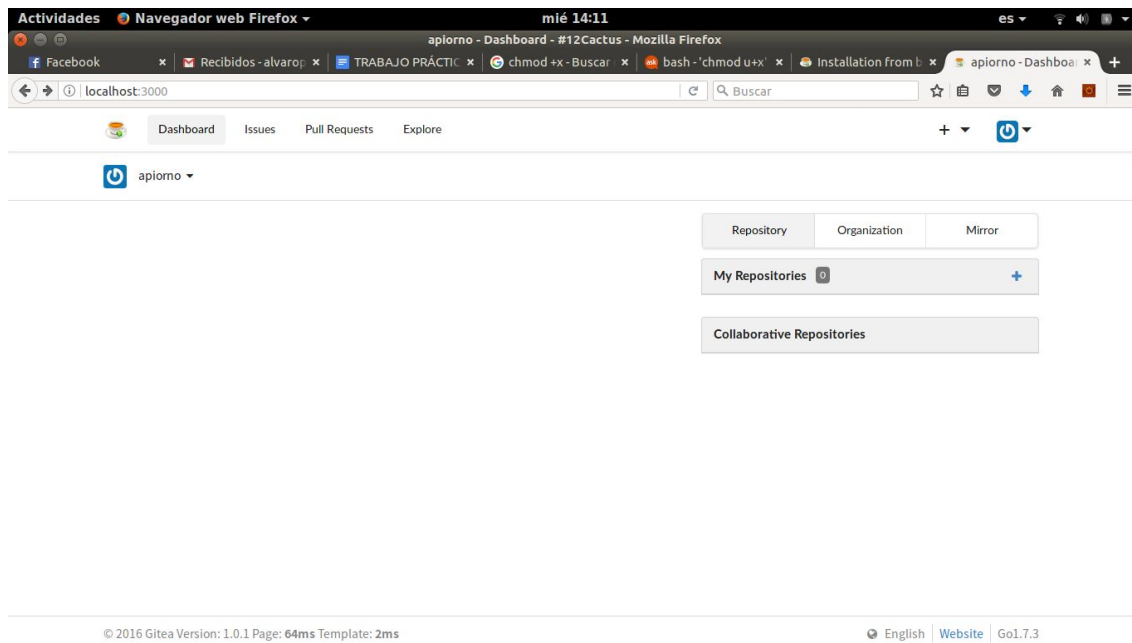
Además de todas estas opciones pueden ser configuradas los envíos de mail y las opciones de administrador que en principio no entraremos en detalle para esta configuración básica.

Para finalizar la instalación hacemos click en *install Gitea*.

Podría aparecer un error "*Database setting is not correct: dial tcp localhost:3300: getsockopt: connection refused*" donde "localhost:3300" es el host elegido, y el problema está relacionado al puerto elegido, el cual podría estar en uso. El 3306 debería funcionar correctamente.

En este punto ya finalizada la instalación lo importante a destacar es que la aplicación trabaja sobre el "localhost:3000" y la base de datos estará hosteada en "localhost:3306". Si alguno está siendo usado por otra aplicación deberá ser modificado por algún otro puerto libre.

Teniendo en claro esto, procedemos a abrir un browser y escribir "localhost:3000" para levantar la aplicación y deberíamos ver algo similar a el siguiente gráfico:



A partir de acá el uso de Gitea es muy similar a los usos de otros proveedores Git como los anteriormente mencionados.