



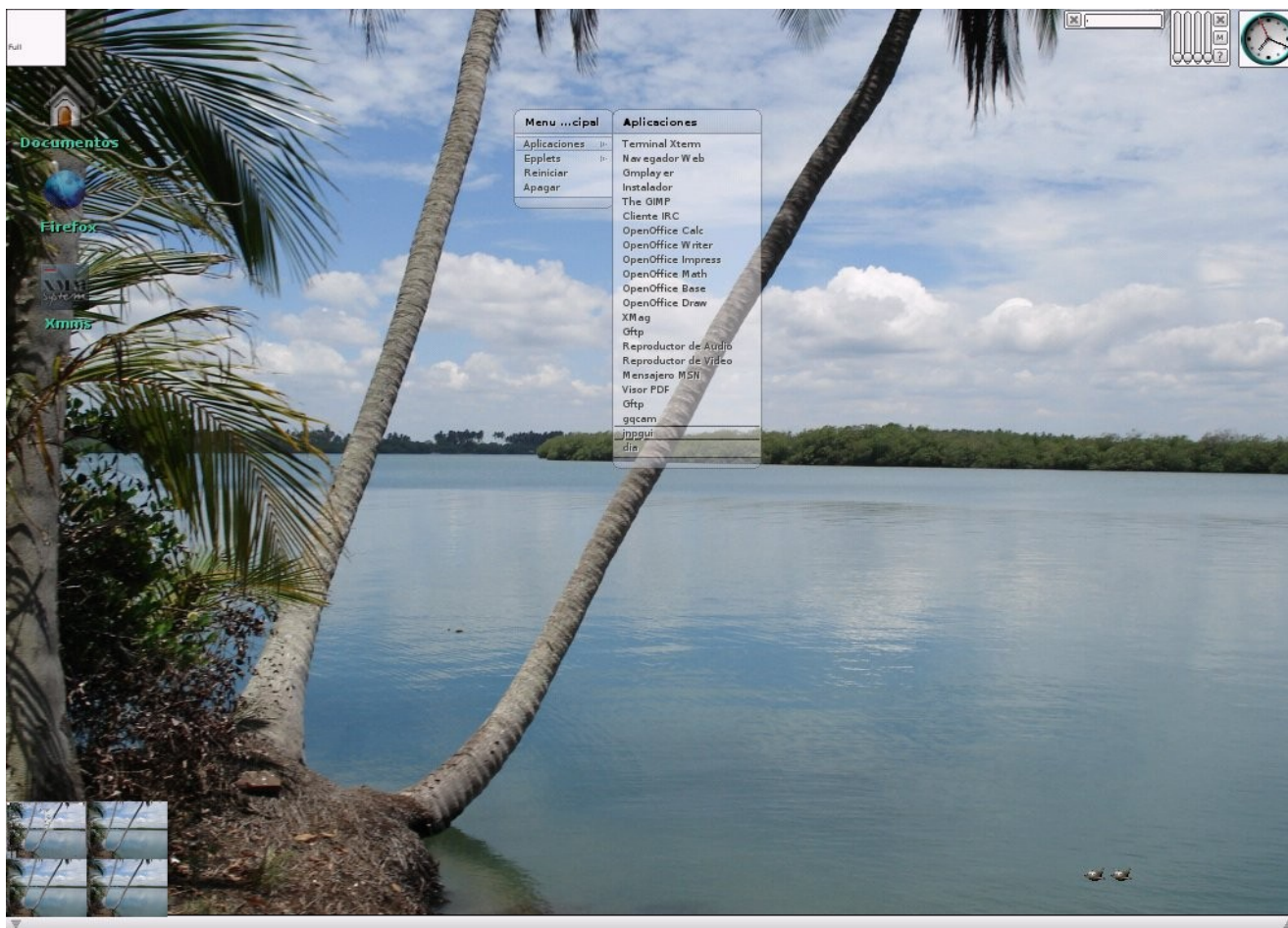
Por: Bachan.
<http://www.el-bulbo.com>

INSTALACION Y CONFIGURACION DE APACHE, MYSQL Y PHP EN JARRONEGRO 2.0.1

En este pequeño tutorial se mostrara como instalar y configurar Apache, Mysql y PHP en JarroNegro ;utilizando jnpgui.

Disculpen las faltas de ortografia (sobre todo los acentos!!).

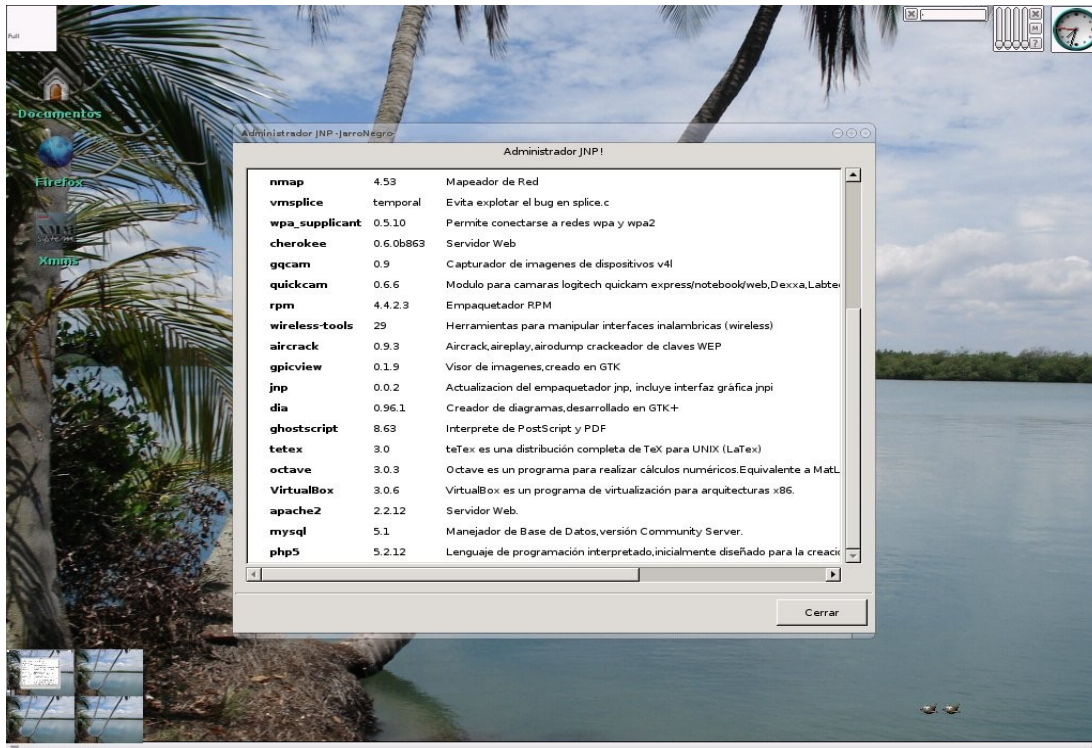
El primer paso es ejecutar el programa jnpgui, damos click con el boton izquierdo en el escritorio,seleccionamos aplicaciones y damos click en jnpgui.



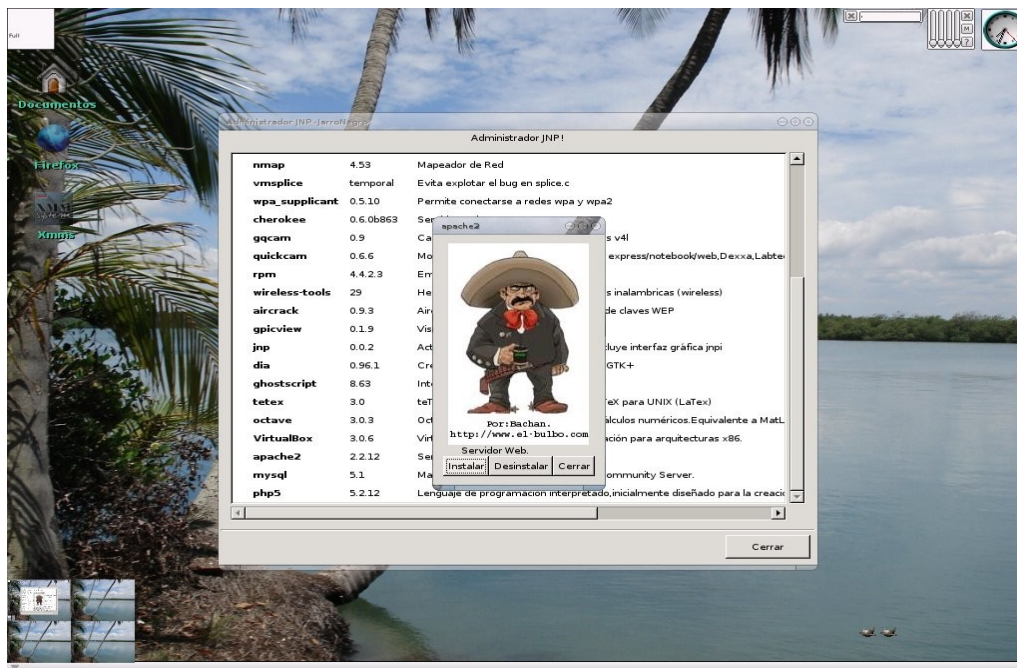
Dependiendo de tu conexion a internet se mostrara una pantalla con los paquetes que existen en nuestros repositorios.

Dudas o comentarios: armando@jarronegrolinux.com

<http://www.jarronegrolinux.com>

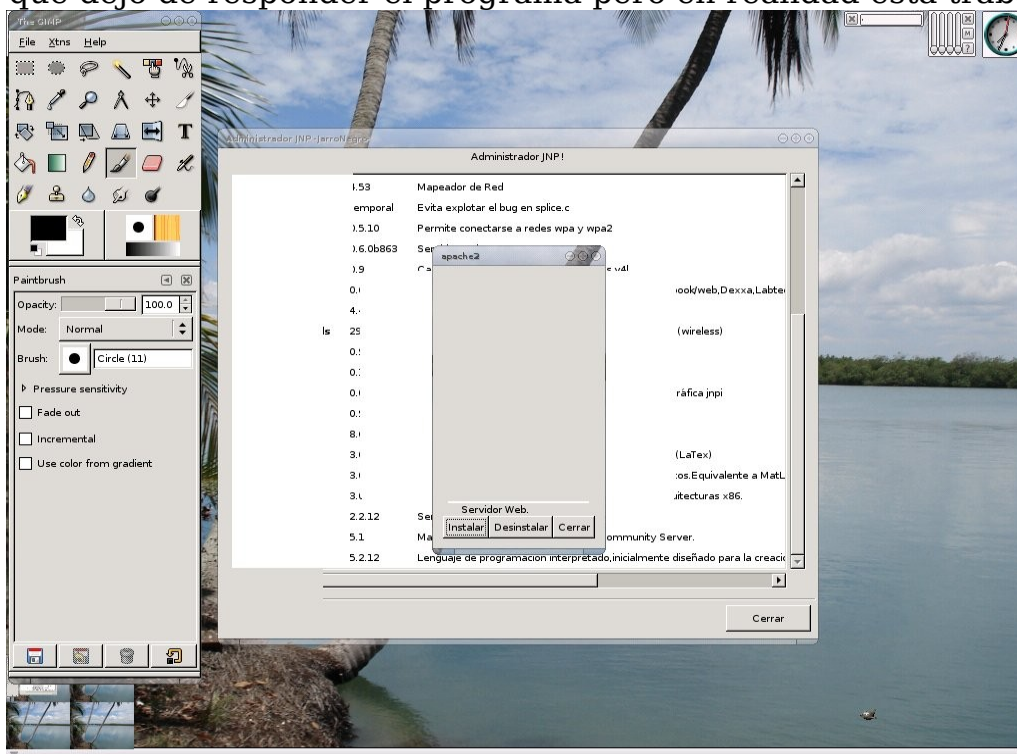


Damos doble click en el paquete apache2, se mostrara una pequeña pantalla como esta:

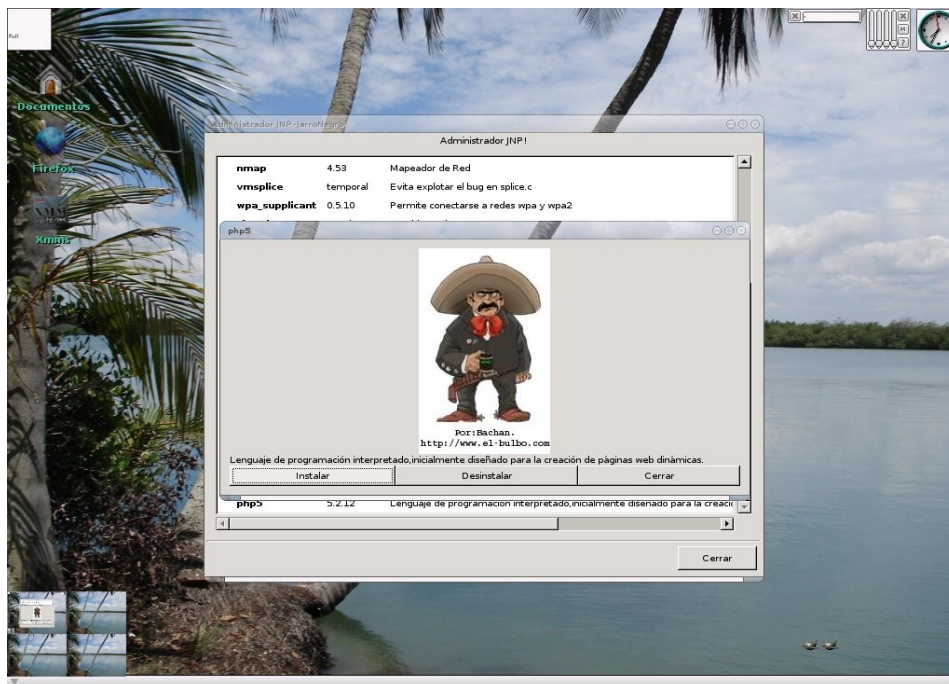


Damos click en instalar y se descargara e instalara el paquete, por el momento el

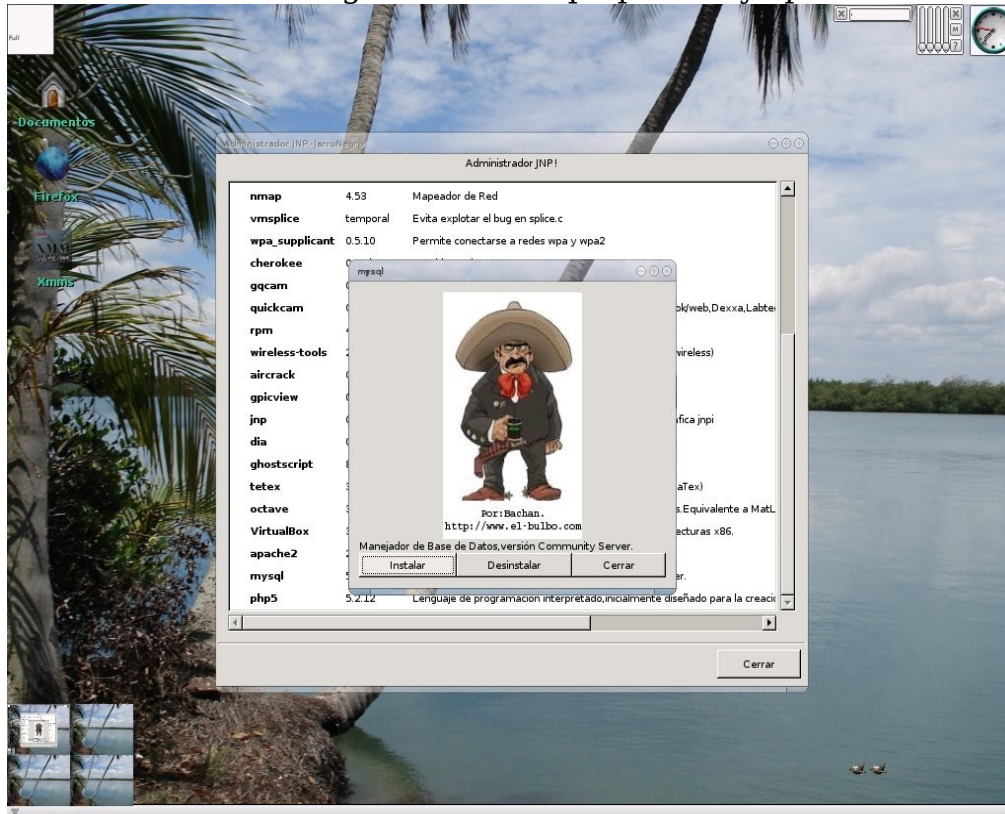
instalador no tiene manera de mostrar el progreso de la descarga, por lo cual pareciera que dejo de responder el programa pero en realidad esta trabajando.



Al terminar la instalacion, el programa jnpgui regresara a su estado normal. Ahora instalamos el paquete php5 de la misma manera..



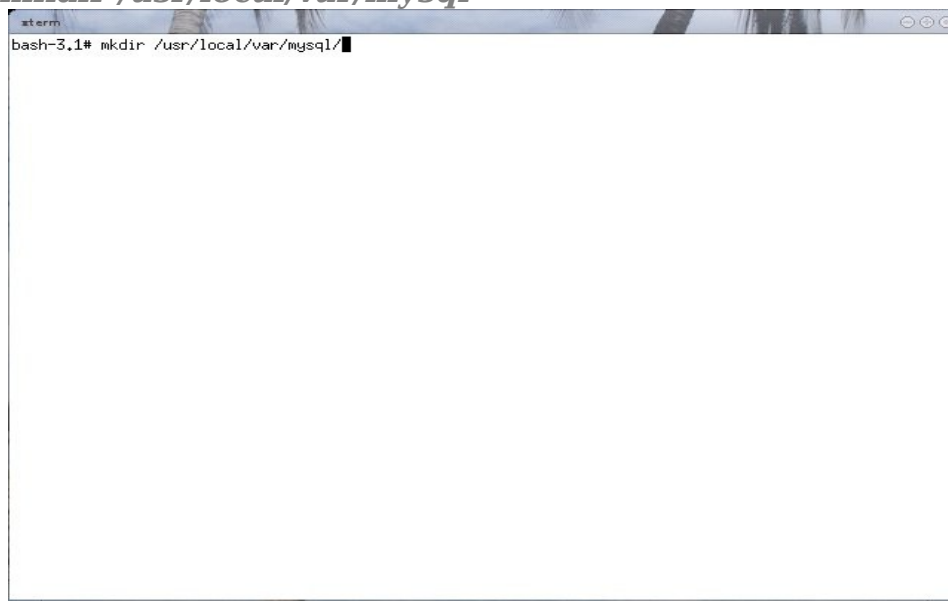
Y finalmente instalaremos de igual forma el paquete mysql!



Ya tenemos instalado todo lo necesario!!

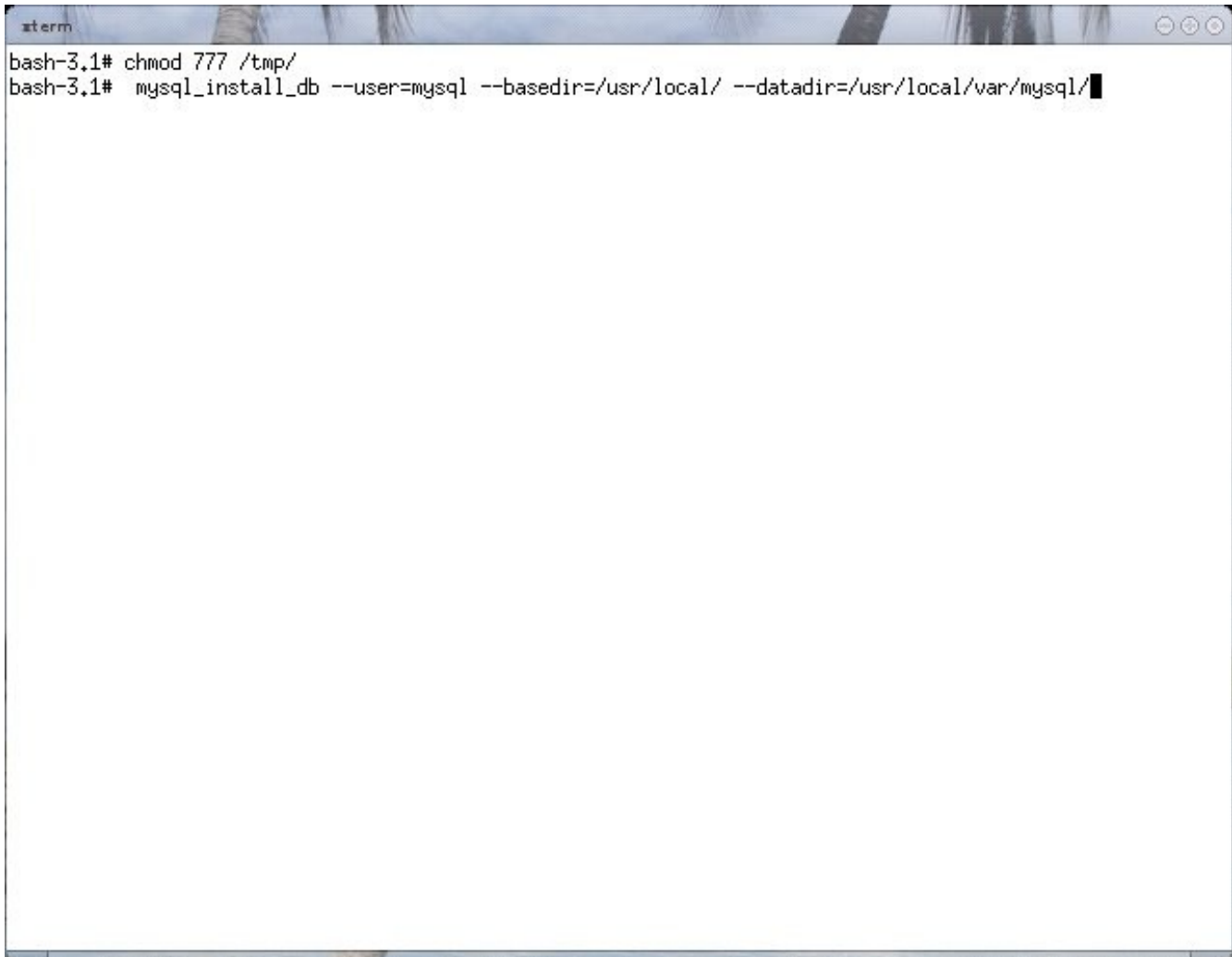
Sin embargo falta configurar los paquetes, especialmente MySQL.

Primero creamos el directorio donde residiran nuestras bases de datos, con el comando: ***mkdir /usr/local/var/mysql***



Para que el script de instalacion de las bases de datos funcione es necesario que el usuario "mysql" tenga permisos para escribir en /tmp ,por lo cual ejecutamos el comando: `chmod 777 /tmp` ,con el cual daremos todos los permisos a todos los usuarios.

Posteriormente ejecutamos el script para instalar las bases de datos para que funcione adecuadamente MySql: `mysql_install_db --user=mysql --basedir=/usr/local/ --datadir=/usr/local/var/mysql/`



```
xterm
bash-3.1# chmod 777 /tmp/
bash-3.1# mysql_install_db --user=mysql --basedir=/usr/local/ --datadir=/usr/local/var/mysql/
```

Al finalizar la instalacion de las bases de datos, se mostrara una pantalla como la siguiente:

```
xterm
bash-3.1# mysql_install_db --user=mysql --basedir=/usr/local/ --datadir=/usr/local/var/mysql/
Installing MySQL system tables...
OK
Filling help tables...
OK

To start mysqld at boot time you have to copy
support-files/mysql.server to the right place for your system

PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER !
To do so, start the server, then issue the following commands:

/usr/local/bin/mysqladmin -u root password 'new-password'
/usr/local/bin/mysqladmin -u root -h JarroNegro password 'new-password'

Alternatively you can run:
/usr/local/bin/mysql_secure_installation

which will also give you the option of removing the test
databases and anonymous user created by default. This is
strongly recommended for production servers.

See the manual for more instructions.

You can start the MySQL daemon with:
cd /usr/local/ ; /usr/local/bin/mysqld_safe &

You can test the MySQL daemon with mysql-test-run.pl
cd /usr/local/mysql-test ; perl mysql-test-run.pl

Please report any problems with the /usr/local/scripts/mysqlbug script!

The latest information about MySQL is available at http://www.mysql.com/
Support MySQL by buying support/licenses from http://shop.mysql.com/

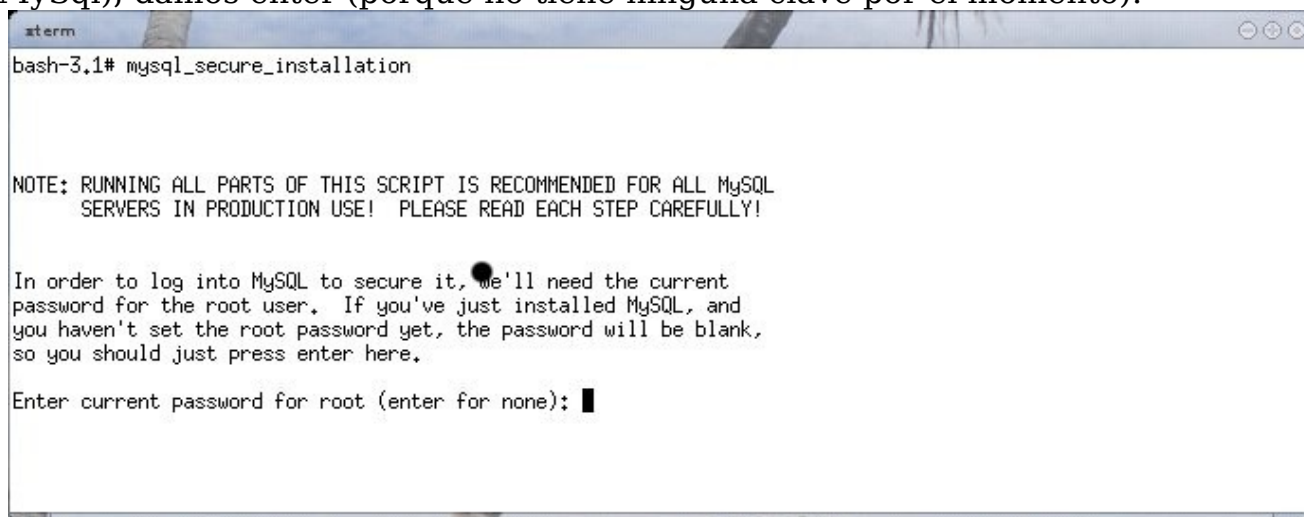
bash-3.1#
```

Ya podemos arrancar el servicio de mysql, para esto ejecutamos el siguiente comando (aunque todavia no es funcional!): `mysqld_safe --basedir=/usr/local --datadir=/usr/local/var/mysql --ledir=/usr/local/libexec --skip-grant-tables`

```
xterm
bash-3.1# mysqld_safe --basedir=/usr/local/ --datadir=/usr/local/var/mysql/ --ledir=/usr/local/libexec/ --skip-gra
nt-tables
091224 08:36:31 mysqld_safe Logging to '/usr/local/var/mysql/JarroNegro.err'.
091224 08:36:31 mysqld_safe Starting mysqld daemon with databases from /usr/local/var/mysql/
█
```

Ahora abrimos otra terminal y ejecutamos el comando: `mysql_secure_installation`

Se mostrara una pantalla como la siguiente, donde nos pide la clave de root (del MySql), damos enter (porque no tiene ninguna clave por el momento):



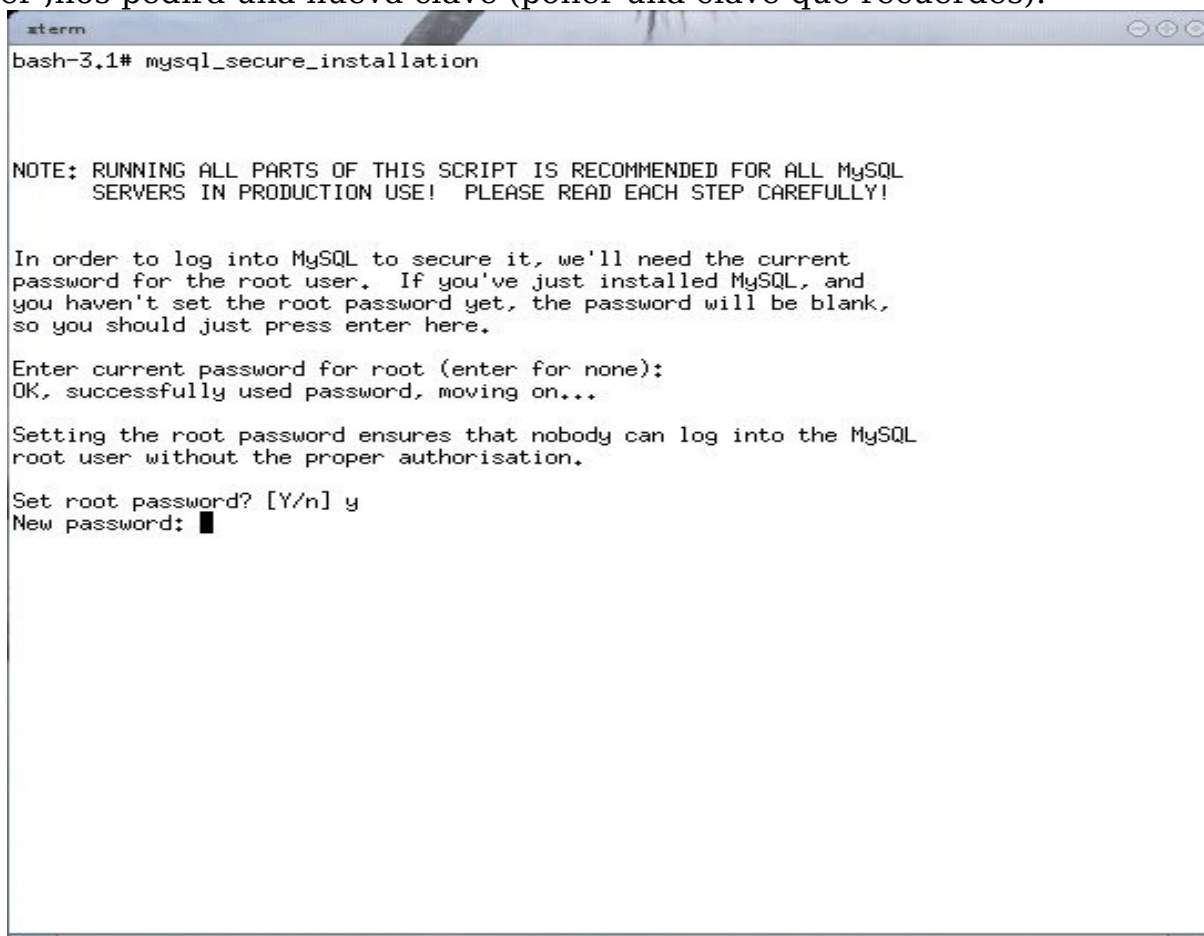
```
xterm
bash-3.1# mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MySQL
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MySQL to secure it, we'll need the current
password for the root user. If you've just installed MySQL, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
```

Despues preguntara si queremos asignarle una clave a root ,le damos Y y enter ,nos pedira una nueva clave (poner una clave que recuerdes).



```
xterm
bash-3.1# mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MySQL
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MySQL to secure it, we'll need the current
password for the root user. If you've just installed MySQL, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MySQL
root user without the proper authorisation.

Set root password? [Y/n] y
New password:
```

El asistente continuara con mas preguntas (podemos darle que si a todas las preguntas o dependiendo de tus necesidades!).

```
xterm
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] y
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y
... Success!

By default, MySQL comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MySQL
installation should now be secure.

Thanks for using MySQL!

bash-3.1# █
```

Ahora podemos probar nuestro manejador de base de datos con el comando:
mysql -u root -p
Nos pedira nuestra contraseña y debe de aparecer una pantalla como esta:


```
zterm
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y
... Success!

By default, MySQL comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MySQL
installation should now be secure.

Thanks for using MySQL!

bash-3.1# mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 5.1.30 Source distribution

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> █
```

Ahora solo nos falta configurar el Apache (esto de acuerdo a sus necesidades, existen muchos tutoriales y libros para configurar estos servicios, por lo cual aqui mostramos de manera muy basica su configuracion e instalacion).

Los paquetes apache2 y php5 incorporan una configuracion muy basica ,por lo cual de ser necesario puede cambiarla de acuerdo a sus necesidades.

El directorio de instalacion del Apache esta en /usr/local/apache2 podemos explorar esta carpeta con el comando: `cd /usr/local/apache2/` y luego con el comando: `ls` vemos el contenido de las carpetas.

```
stern
bash-3.1# cd /usr/local/apache2/
bash-3.1# ls
bin  build  cgi-bin  conf  error  htdocs  icons  include  lib  logs  man  manual  modules
bash-3.1#
```

Ahora para probar nuestra instalacion de Apache, Mysql y PHP creamos la tipica pagina "hola mundo" en PHP.

Para esto nos vamos al directorio `/usr/local/apache2/htdocs` con el comando: `cd /usr/local/apache2/htdocs`

y creamos una pagina con el comando: `nano index.php`

```
stern
bash-3.1# cd htdocs/
bash-3.1# ls
apache_pb.gif  apache_pb22.gif  apache_pb22_ani.gif  index.php
apache_pb.png  apache_pb22.png  index.html
bash-3.1# nano index.php
```

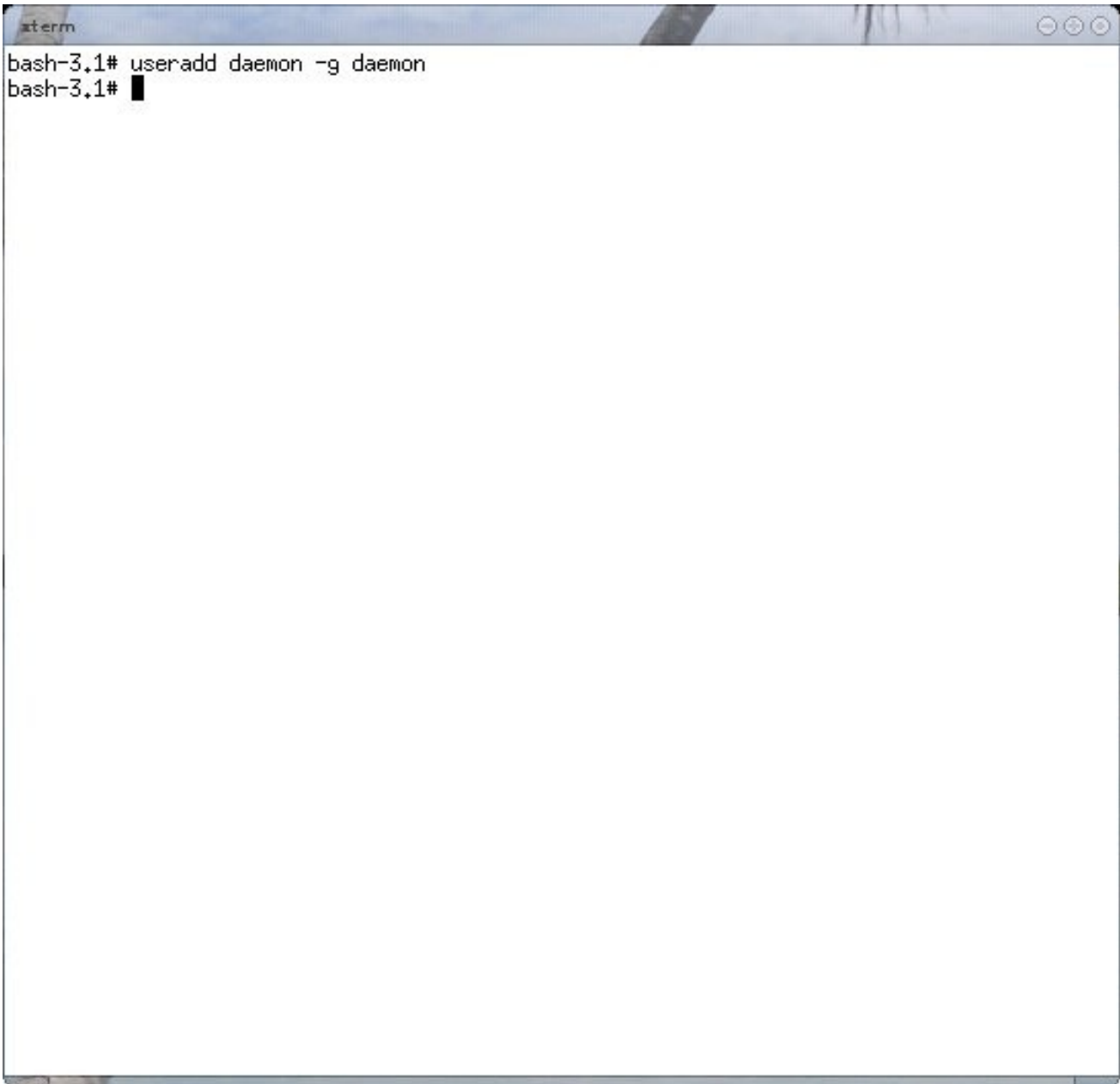
Y ponemos el contenido como el que sigue o puedes sacar al programador web que tienes en tu interior!



```
xterm
GNU nano 2.0.1 File: index.php
?php
echo "HOLA";
?>

[ Read 4 lines ]
^G Get Help    ^O WriteOut    ^R Read File   ^Y Prev Page   ^K Cut Text    ^C Cur Pos
^X Exit        ^J Justify     ^W Where Is    ^V Next Page   ^U UnCut Text  ^T To Spell
```

Ahora necesitamos crear al usuario “daemon” para que se ejecute correctamente el Apache con el comando: `useradd daemon -g daemon`

A terminal window titled 'xterm' with standard window controls. The prompt is 'bash-3.1#'. The command 'useradd daemon -g daemon' has been entered and executed. The prompt is now 'bash-3.1#' with a cursor. The rest of the terminal is empty.

```
xterm
bash-3.1# useradd daemon -g daemon
bash-3.1#
```

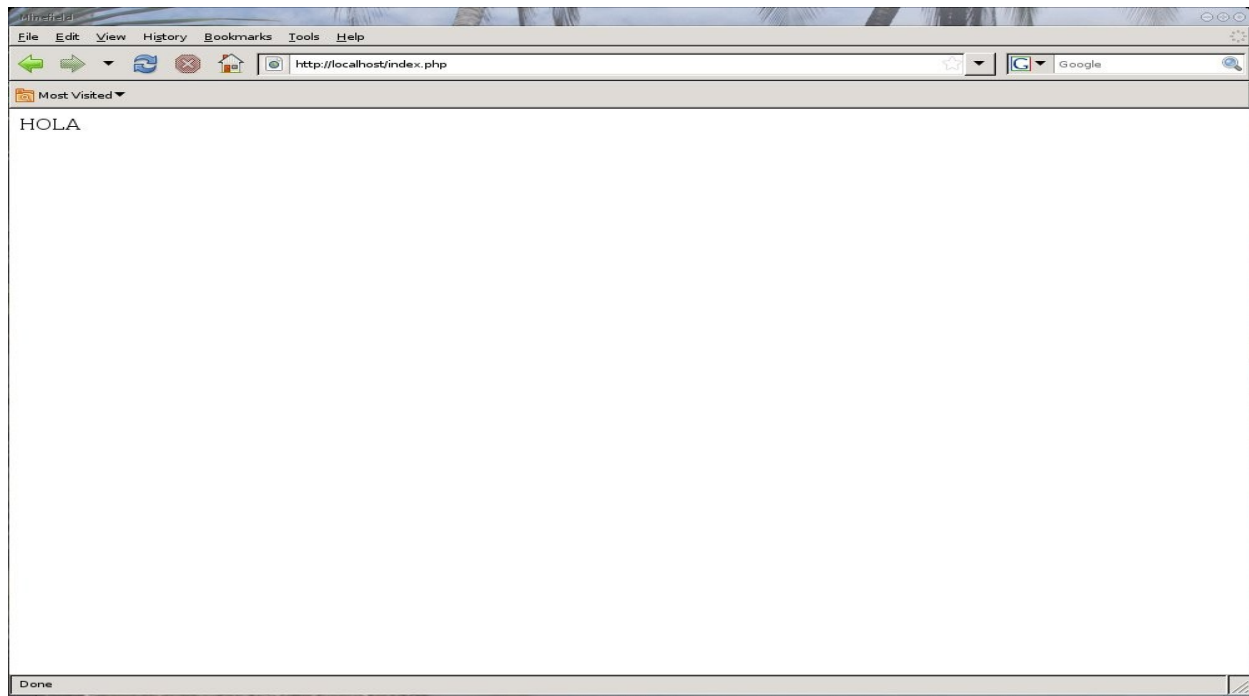
Ahora ejecutamos el apache con el comando: `/usr/local/apache2/bin/httpd`
Y podemos ver si se estan ejecutando todos nuestros servidores con el comando:
`nmmap localhost` y se mostrara una imagen como la siguiente.

```
xterm
bash-3.1# useradd daemon -g daemon
bash-3.1# /usr/local/apache2/bin/httpd
[Thu Dec 24 10:51:46 2009] [warn] module php5_module is already loaded, skipping
[Thu Dec 24 10:51:46 2009] [warn] module php5_module is already loaded, skipping
bash-3.1# nmap localhost

Starting Nmap 4.53 ( http://insecure.org ) at 2009-12-24 10:51 UTC
Unable to find nmap-services! Resorting to /etc/services
Interesting ports on jarrito.example.org (127.0.0.1):
Not shown: 5069 closed ports
PORT      STATE SERVICE
80/tcp    open  http
631/tcp   open  ipp
3306/tcp  open  mysql

Nmap done: 1 IP address (1 host up) scanned in 1.803 seconds
bash-3.1# █
```

Para finalizar abrimos el navegador web Firefox y abrimos la pagina <http://localhost/index.php> y se mostrara nuestra pagina que creamos anteriormente!



Ya tenemos listo nuestro servidor con Apache Mysql y PHP !!
Ahora podemos instalar cualquier aplicacion web o desarrollarla!

Cualquier duda o comentario, pueden escribirme a mi correo electronico:
armando889@yahoo.com o armando@jarronegrolinux.com

Saludos a toda la comunidad!!

¡¡Feliz Navidad!!
Armando Rodriguez Arguijo
24/12/2009