

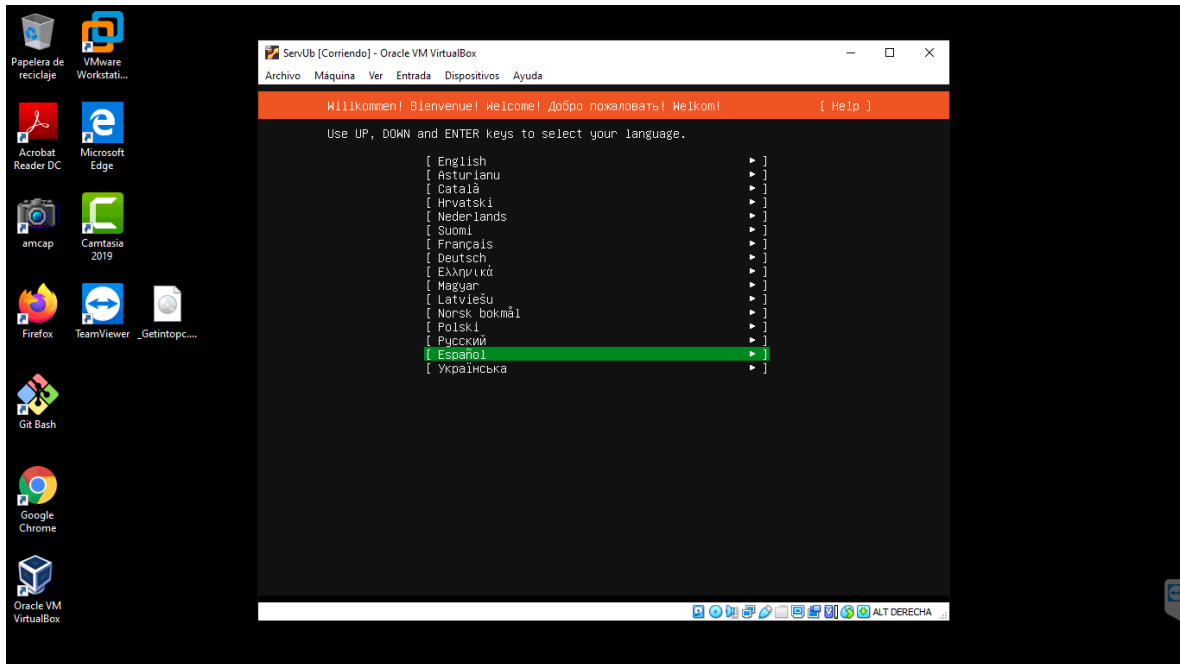
Proyecto: Implementación de Sistema de Monitoreo para cultivo protegido con mejora de disponibilidad y reducción en volumen de datos.

Por: Orlando David Orbes.

Procedimiento: Instalación de Servidor Ubuntu IoT en máquina virtual

Ajustar configuración de Virtualbox para instalación de Ubuntu 64 bits, disco duro 20Gb y red a modo puente.

Selección idioma y de teclado.



Conexiones de red

Reporta DHCPv4 192.168.1.26/24, en razón de conectarse por la interfaz inalámbrica, así como solicita dirección proxy de requerir para la conexión. Establece como mirror address: <http://co.archive.ubuntu.com/ubuntu>

Usuario: Orlando Orbes

Servidor: labserver

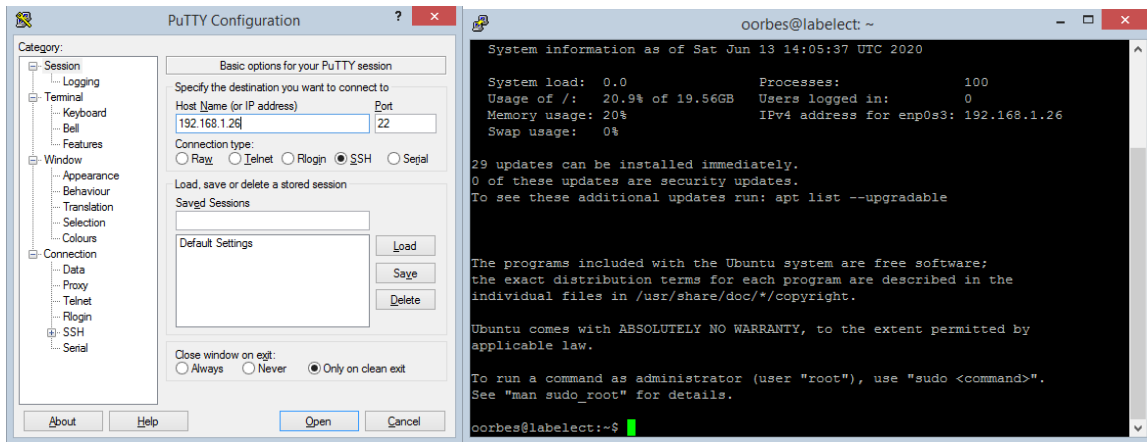
username: oorbes

password: ordag2000.

Se habilita la opción de instalar OpenSSH server

Asegurarse de activar la máquina virtual con interfaz de red en bridge.

Para el ingreso se utiliza putty, dando la dirección asignada dinámicamente por el router al servidor.



1. Creación de usuario con privilegios

Antes de continuar crearé una cuenta de usuario adicional con privilegios de superusuario:

\$sudo adduser orlando	Adicionar usuario
\$sudo usermod -aG sudo orlando	Agregar al grupo superusuarios
\$sudo su - orlando	Sustituir usuario actual a orlando.

Para evitar que se solicite permanentemente contraseña, puede usarse el comando

#sudo su

3. Configuración de firewall: Bloqueo de conexiones.

La instalación de firewall bloqueará todas las conexiones, excepto SSH. El comando usado es ufw con las opciones allow seguido de la aplicación habilitada. Se uso también app list para listar aplicaciones, enable para habilitar el servicio y status para conocer los atributos asociados a las aplicaciones.

\$sudo ufw allow OpenSSH

\$sudo ufw enable

\$sudo ufw status

4. Revisión de la zona de tiempo

Se verifica el registro de fecha y hora con \$date

Se procede al ajuste usando:

\$sudo timedatectl set-timezone America/Bogota

5. Actualización de Ubuntu

```
$ sudo apt update && sudo apt upgrade && sudo apt dist-upgrade
```

```
$ sudo reboot
```

6. Crear espacio de intercambio Swap, para evitar bloqueo cuando haya carga pesada.

Verificación de configuración previa:

```
$swapon --show
```

Se reporta su configuración y a través del comando free se verifica su uso actual:

```
$free -h
```

7. Instalación de VNC

- Instalación Xfce y VNC

```
$sudo apt update
```

```
$sudo apt install xfce4 xfce4-goodies tightvncserver
```

Configuración del servidor VNC

```
$vncserver
```

Requiere un password de 8 caracteres: servncoo

Es necesario editar el archivo de configuración para lo cual:

```
$vncserver -kill :1
```

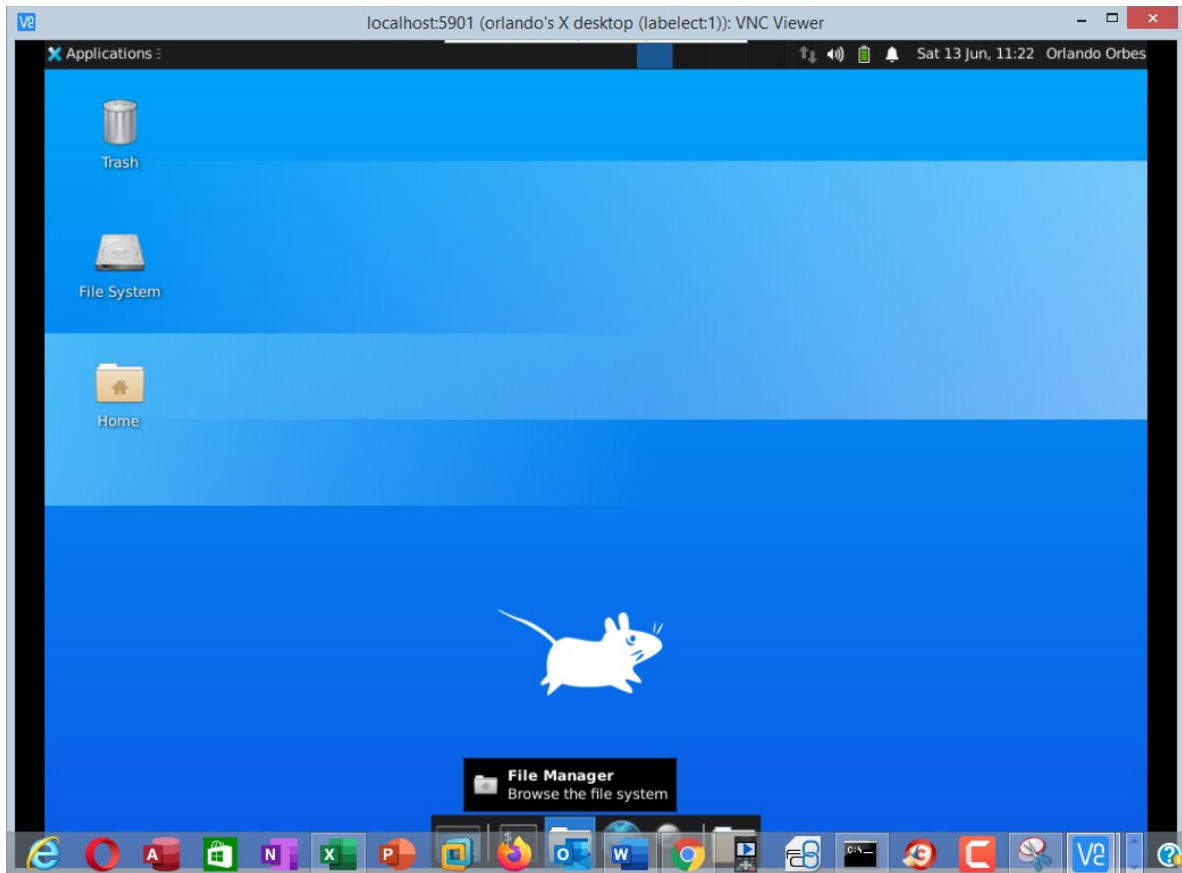
Crear una copia y edita el archivo xstartup agregando al final del archivo "startxfce4 &" a fin de que se active el ambiente de escritorio cuando inicie el servidor VNC. Finalmente se cambian las propiedades del archivo para hacerlo ejecutable.

El detalle se encuentra en el enlace: <https://devanswers.co/install-vnc-ubuntu-18-04-windows-mac-linux/>

Crear un túnel SSH, habilitando comunicaciones encriptadas.

En el servidor se usa el comando:

```
$ ssh -L 5901:127.0.0.1:5901 -C -N -l orlando 192.168.1.26
```



Ejecución de VNC con un servicio

Instalación del servidor Web Apache.

Actualiza el administrador de paquetes.

```
#apt update
```

```
#apt install apache2
```

Para permitir el tráfico de este servidor es necesario agregarlo a la lista del firewall, se procede con estos comandos:

- Despliegue de aplicaciones disponibles:

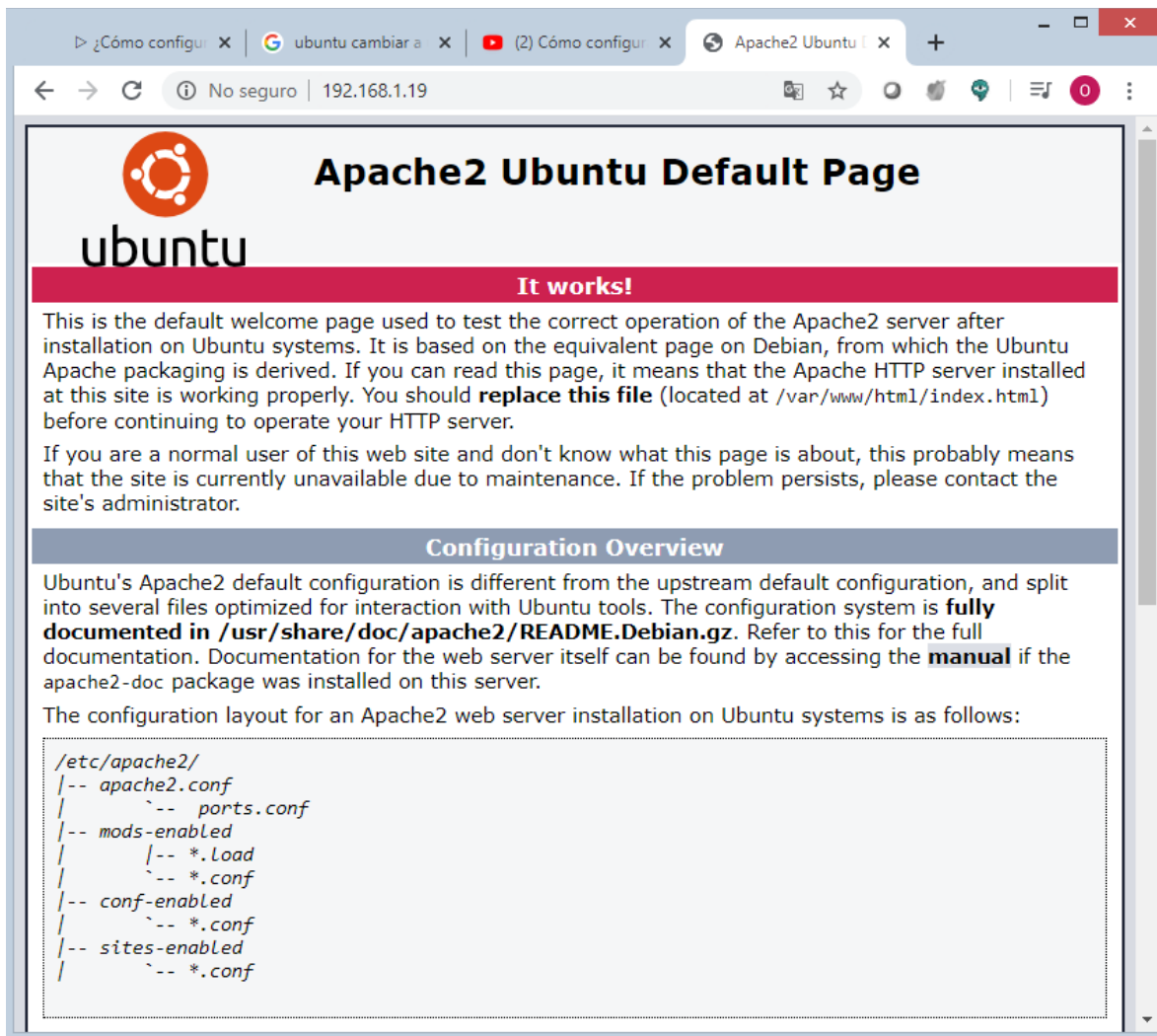
```
#ufw app list
```

```
#sudo ufw app info "Apache Full"
```

```
#sudo ufw allow in "Apache Full"
```

Con esto se ha verificado que Apache permite el tráfico a través de los puertos HTTP (80) y HTTPS (443).

La verificación del servicio se puede verificar desde un navegador con la dirección del servidor.



Instalación de MySQL, base de datos del sistema.

```
#apt install mysql-server
```

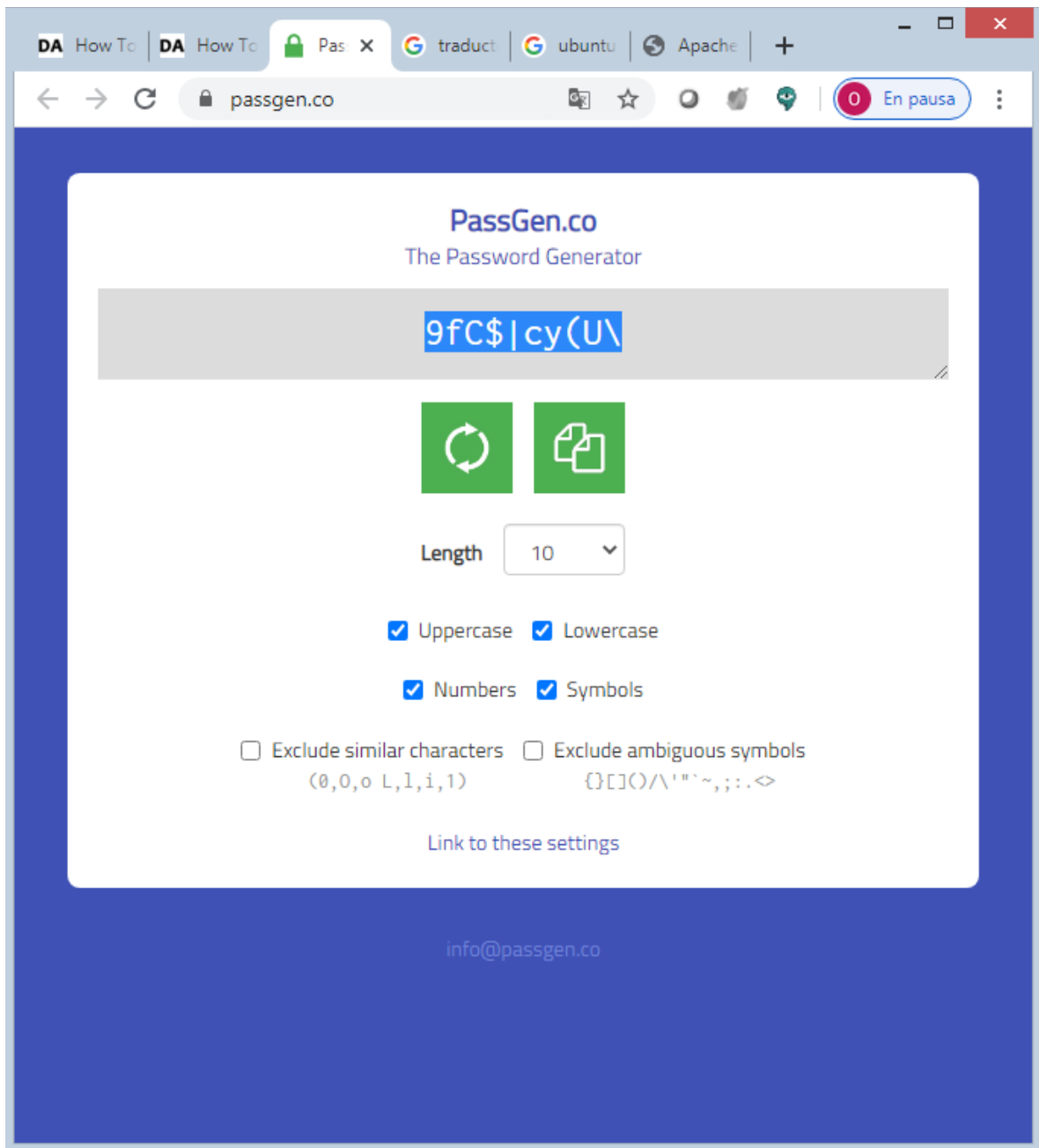
Para verificar que la instalación es correcta, se ejecuta el script preinstalado para ayudarnos a verificar el sistema de base de datos.

```
#mysql_secure_installation
```

El componente de validación opcional de contraseñas, no es instalado debido a problemas encontrados en la primera instalación. A partir de aquí se siguen orientaciones del enlace:

<https://devanswers.co/install-apache-mysql-php-lamp-stack-ubuntu-20-04/>

5.2 Creación contraseña para el root.



Usando los generadores de contraseñas fuertes, se usa la siguiente: 9fC\$|cy(U/ Se cambia el último símbolo.

Se continúan con los demás pasos como eliminar usuarios anónimos y la base de datos de prueba. También deshabilita los registros root remotos y carga estas nuevas reglas así que MySQL inmediatamente fuerza los cambios que hemos hecho.

~~Es necesario definir el mecanismo de autenticación para el usuario root, por lo que se cambiará con los siguientes comandos:~~

```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY  
'Orlando2020.';
```

```
mysql> FLUSH PRIVILEGES;
```

```
mysql> SELECT user,authentication_string,plugin,host FROM mysql.user WHERE user=root;
```

```
mysql> exit
```

Con estos comandos se cambia y verifica en ajuste.

5.7 Prueba del servicio de MySQL

\$sudo mysql – Se verifica la habilitación del entorno.

6. Instalación de PHP

Instalación de PHP o preprocesador de Hipertexto, un lenguaje de script para servidores usado para desarrollo de aplicaciones web y servicios.

```
sudo apt update && sudo apt install php libapache2-mod-php php-mysql
```

y la verificación:

```
php -version
```

Prueba PHP para Apache

Se crea el archivo info.php, en el directorio raíz.

```
sudo nano /var/www/html/info.php
```

agregando el código

```
<?php  
phpinfo();
```

Se graba y cierra, luego se prueba con la dirección del servidor seguido del nombre del archivo
info.php: <http://192.168.1.26/info.php>

PHP Version 7.4.3	
System	Linux labelct 5.4.0-37-generic #41-Ubuntu SMP Wed Jun 3 18:57:02 UTC 2020 x86_64
Build Date	May 26 2020 12:24:22
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.4/apache2
Loaded Configuration File	/etc/php/7.4/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.4/apache2/conf.d
Additional .ini files parsed	/etc/php/7.4/apache2/conf.d/10-mysqld.ini, /etc/php/7.4/apache2/conf.d/10-opcache.ini, /etc/php/7.4/apache2/conf.d/10-pdo.ini, /etc/php/7.4/apache2/conf.d/20-calendar.ini, /etc/php/7.4/apache2/conf.d/20-ctype.ini, /etc/php/7.4/apache2/conf.d/20-exif.ini, /etc/php/7.4/apache2/conf.d/20-ffi.ini, /etc/php/7.4/apache2/conf.d/20-fileinfo.ini, /etc/php/7.4/apache2/conf.d/20-ftp.ini, /etc/php/7.4/apache2/conf.d/20-gettext.ini, /etc/php/7.4/apache2/conf.d/20-iconv.ini, /etc/php/7.4/apache2/conf.d/20-json.ini, /etc/php/7.4/apache2/conf.d/20-mysqli.ini, /etc/php/7.4/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.4/apache2/conf.d/20-phar.ini, /etc/php/7.4/apache2/conf.d/20-posix.ini, /etc/php/7.4/apache2/conf.d/20-readline.ini, /etc/php/7.4/apache2/conf.d/20-shmop.ini, /etc/php/7.4/apache2/conf.d/20-sockets.ini, /etc/php/7.4/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.4/apache2/conf.d/20-sysvsem.ini, /etc/php/7.4/apache2/conf.d/20-sysvshm.ini, /etc/php/7.4/apache2/conf.d/20-tokenizer.ini
PHP API	20190902
PHP Extension	20190902
Zend Extension	320190902
Zend Extension Build	API320190902.NTS
PHP Extension Build	API20190902.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled

El archivo se elimina para no dejar abierta información reservada.

Tener en cuenta la nota:

Note: If you plan on uploading files larger than 2MBs through WordPress or similar, you will need to alter the PHP config file and set the max upload size. See: [PHP / Apache: set max file upload and post size](#).

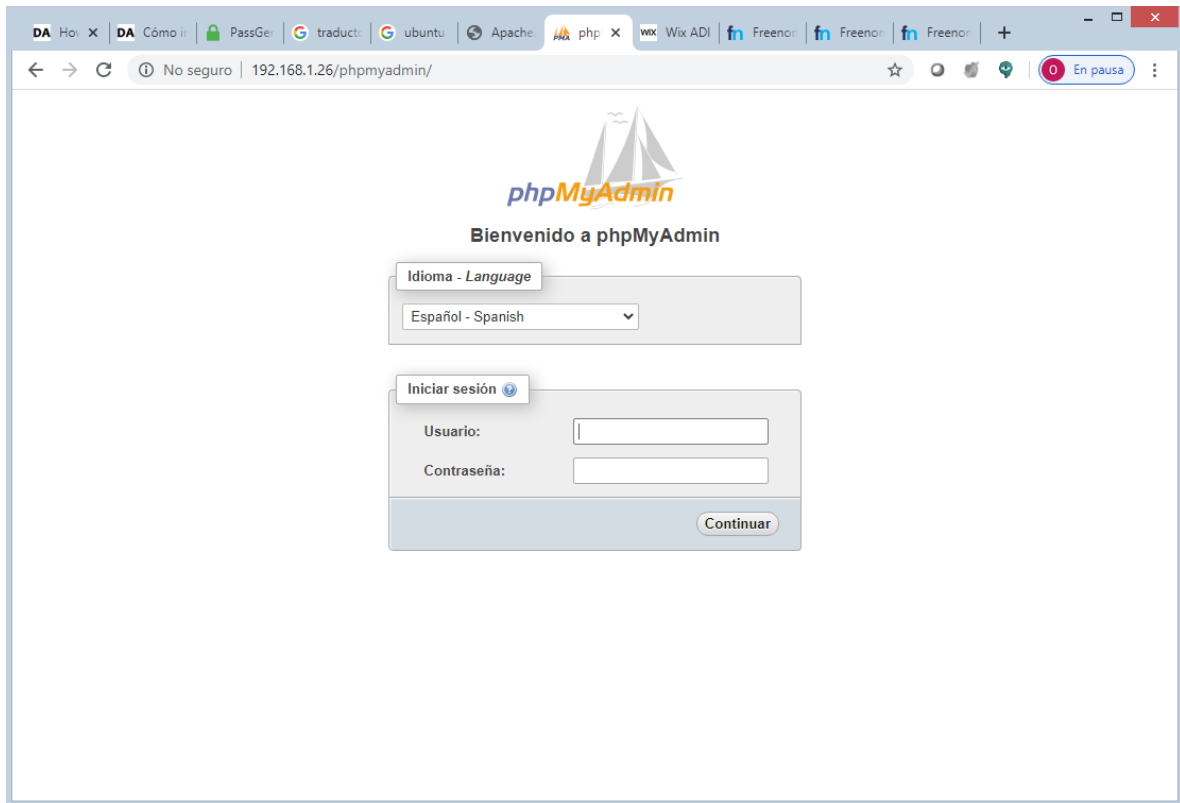
Instalación de phpMyAdmin

```
sudo apt update && sudo apt install phpmyadmin php-mbstring
```

Durante el proceso se selecciona Apache2

A la solicitud del password para phpadmin para su registro con el servidor de base de datos se deja en blanco para que sea automáticamente generado.

Prueba de phpMyAdmin



A continuación se creará un superusuario para esta administración

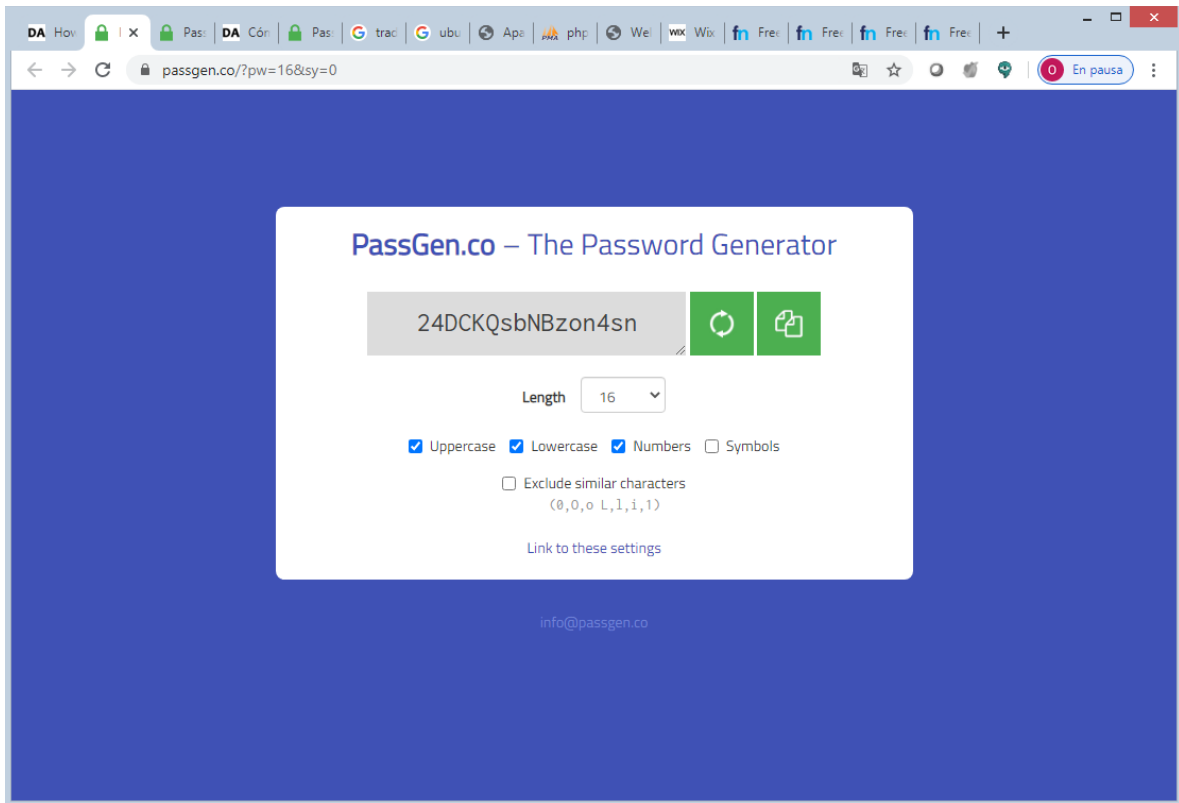
```
sudo mysql -u root -p
```

Es necesaria una contraseña fuerte, por lo que es generada 1v/uY\9!l,

```
CREATE USER 'pmauser'@'localhost' IDENTIFIED BY ' 1v/uY\9!l,'; No funciona
```

Usuario php: phppadmin Contraseña: r0a438\Hp7 ; No funciona

Se actualiza la contraseña para el root así:



24DCKQsbNBzon4sn

Superusuario: phpad Contraseña: as32<09BF,

INSTALACIÓN DE NODE.JS Y NODE-RED

Sin novedad.

INSTALACION del corredor de mensajes

Message Broker

#apt update

#apt install mosquitto mosquitto-clients

Prueba:

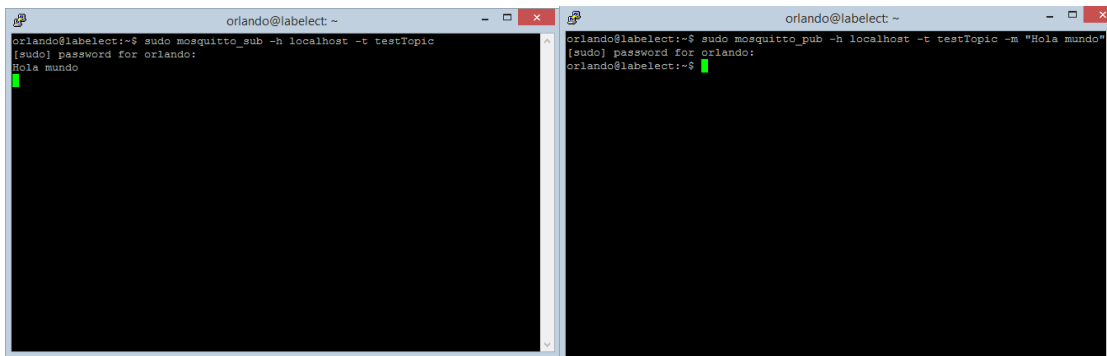
lsof -i :1883 Responde con procesos de escucha en el puerto 1883.

Se verifica el protocolo de publicación-suscripción a través de comandos de prueba:

#mosquitto_sub -h localhost -t testTopic

Con -h se especifica el nombre del host, y con -t el tópico de suscripción. Ejecutado en un terminal espera escuchando. Desde otro terminal se dirige la publicación así:

#mosquitto_pub -h localhost -t testTopic -m "Hola"



Seguridad para el corredor de mensajes

#mosquitto_passwd -c /etc/mosquitto/passwd anand Crea nuevo archive de contraseñas.

Para este primer caso la contraseña es dispcom1

#mosquitto_passwd /etc/mosquitto/passwd guest

Para este primer caso la contraseña es dispcom2

Esto tiene efecto con la edición del archivo de configuración y reinicio del bróker

#nano /etc/mosquitto/conf.d/broker.conf

En este archivo de configuración se evita el acceso anónimo y se indica el archivo de contraseñas:

Allow_anonymous false

password_file /etc/mosquitto/passwd

A continuación se cierran todos los procesos

lsof -i : 1883 Identifica procesos asociados

kill num_proceso

El reinicio se da con la orden

#mosquitto -c /etc/mosquitto/conf.d/broker.conf -v &

Para completar la configuración del bróker se agregan líneas al archivo de configuración asociadas con habilitación del registro de nuevas conexiones, subscripciones, publicaciones, desconexiones, errores y advertencias. Intervalo de registro de 15 minutos, persistencia de las sesiones, acceso mediante autenticación, y para incrementar la seguridad cambio de puerto por defecto y uso de archivos de certificación SSL. Lo último se descarta hasta tener el nombre de dominio.

Con la siguiente orden se reinicia el broker luego se cierran procesos activos, permitiendo su ejecución y registro de actividades en el archivo .log.

mosquitto -c /etc/mosquitto/conf.d/broker.conf -v > mosquitto.log &

CONSTRUCCION DE COMPONENTES CRITICOS

Base de datos núcleo de la serie de tiempo

Usando phpMyAdmin se creará la base de datos tSeriesDB

Se requiere una nueva cuenta de usuario, se crea así

Usuario cloudadm

Contraseña cloud20admin.

INSTALACION ELEMENTOS NODE

Se genera el password hash mediante el comando:

node-red-admin hash-pw

Introduciendo la contraseña de node-red-admin, Nod20red+

Se obtiene el código encriptado

```
orlando@labelect:/home$ sudo npm install -g node-red-admin
/usr/local/bin/node-red-admin -> /usr/local/lib/node_modules/node-red-admin/node
-red-admin.js
+ node-red-admin@0.2.6
updated 1 package in 4.589s
orlando@labelect:/home$ node-red-admin hash-pw
Password:
$2a$08$eJd/3W7ctt7gtX/FgNa2vuM9/VnROG7apvWRVnSHpE0EVWY5IKa3S
```

Desde una segunda consola y ejecutando node-red se accede al archivo settings.js y se habilita el ingreso mediante contraseña, que antes estaba abierto.

```
// Securing Node-RED
// -----
// To password protect the Node-RED editor and admin API, the following
// property can be used. See http://nodered.org/docs/security.html for details.
adminAuth: {
  type: "credentials",
  users: [{
    username: "admin",
    password: "YOUR_PASSWORD_HERE",
    permissions: "*"
  }]
},
```

Si se necesita jugar con otras características de Node-red, puede consultarse la página

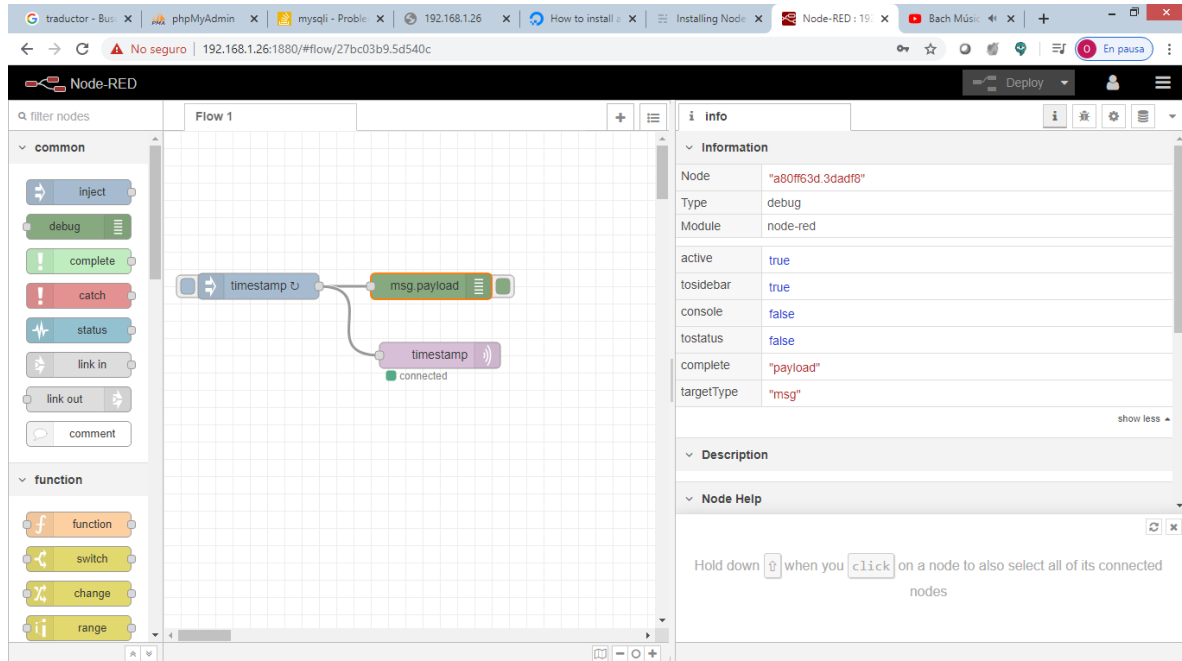
<http://richardn.ca/2019/07/03/installing-node-red-on-ubuntu-2/>

Se ingresa desde navegador a node-red usando el usuario y contraseña establecidos previamente

Usuario: admin Contraseña: Nod20red+

A continuación se agrega un nodo para acceso a mysql, desde el administrador de paletas. Paso siguiente se procede a crear el primer flujo usando un nodo inject, otro de depuración y uno de mqtt out, a fin de crear un escenario de publicación de mensajes con periodicidad de 15 segundos.

El flujo obtenido se aprecia:



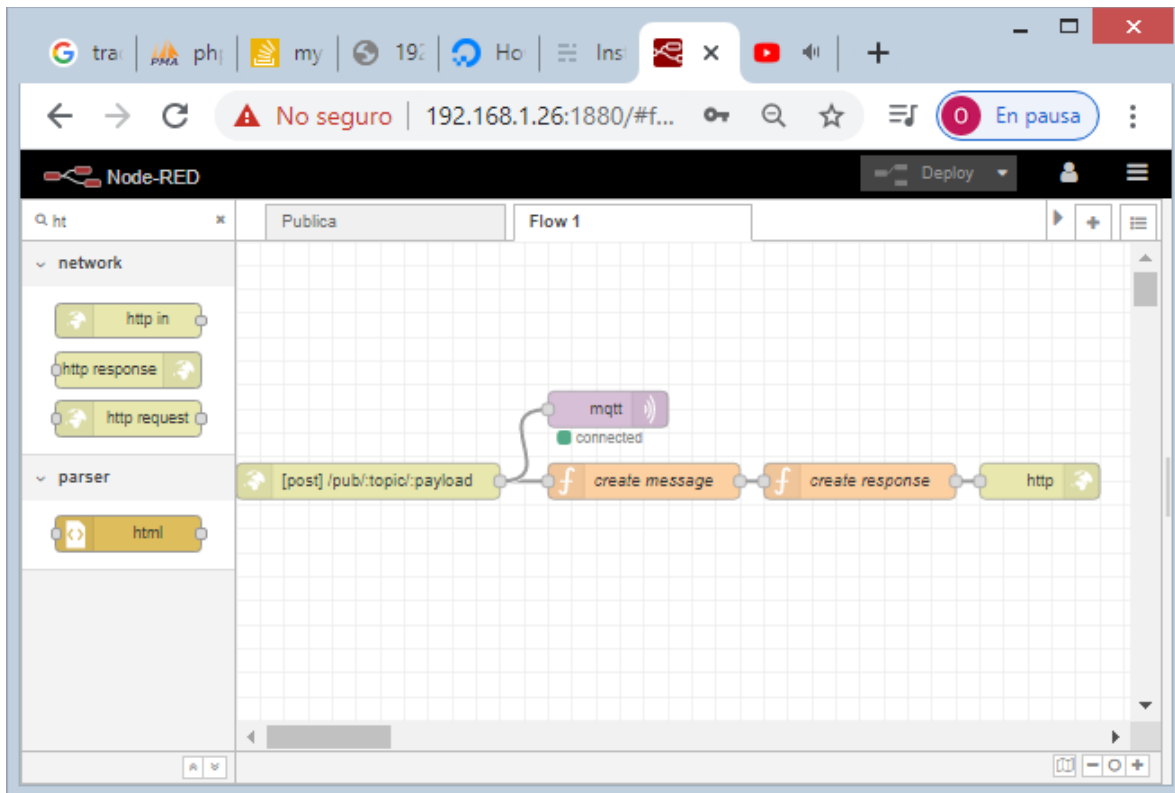
```

orlando@labelect: ~
orlando@labelect:~$ sudo nano /home/orlando/.node-red/settings.js
orlando@labelect:~$ sudo nano /home/orlando/.node-red/settings.js
[sudo] password for orlando:
orlando@labelect:~$ sudo lsof -i: 1883
[sudo] password for orlando:
lsof: unacceptable port specification in: -i :
lsof 4.93.2
latest revision: https://github.com/lsof-org/lsof
latest FAQ: https://github.com/lsof-org/lsof/blob/master/00FAQ
latest (non-formatted) man page: https://github.com/lsof-org/lsof/blob/master/L
ssof.8
usage: [-?abhKlnNoOPRtUvVX] [+|-c c] [+|-d s] [+D D] [+|-E] [+|-e s] [+|-f[gG]]
[-F [f]] [-g [s]] [-i [i]] [+|-L [l]] [+m [m]] [+|-M] [-o [o]] [-p s]
[+|-r [t]] [-s [p:s]] [-S [t]] [-T [t]] [-u s] [+|-w] [-x [fl]] [--] [names]
Use the ``-h'' option to get more help information.
orlando@labelect:~$ sudo lsof -i :1883
COMMAND      PID      USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
mosquitto    3970    mosquitto   5u  IPv4  72266      0t0  TCP *:1883 (LISTEN)
mosquitto    3970    mosquitto   6u  IPv6  72267      0t0  TCP *:1883 (LISTEN)
mosquitto    3970    mosquitto   7u  IPv4 115141      0t0  TCP localhost:1883->localh
ost:44506 (ESTABLISHED)
node-red     6223     orlando   22u  IPv4 115140      0t0  TCP localhost:44506->local
host:1883 (ESTABLISHED)
orlando@labelect:~$

```

Se aprecia en la ventana de depuración, el estado activo del flujo y en el terminal el reporte de procesos activos del protocolo mqtt

Publicador de mensajes REST API

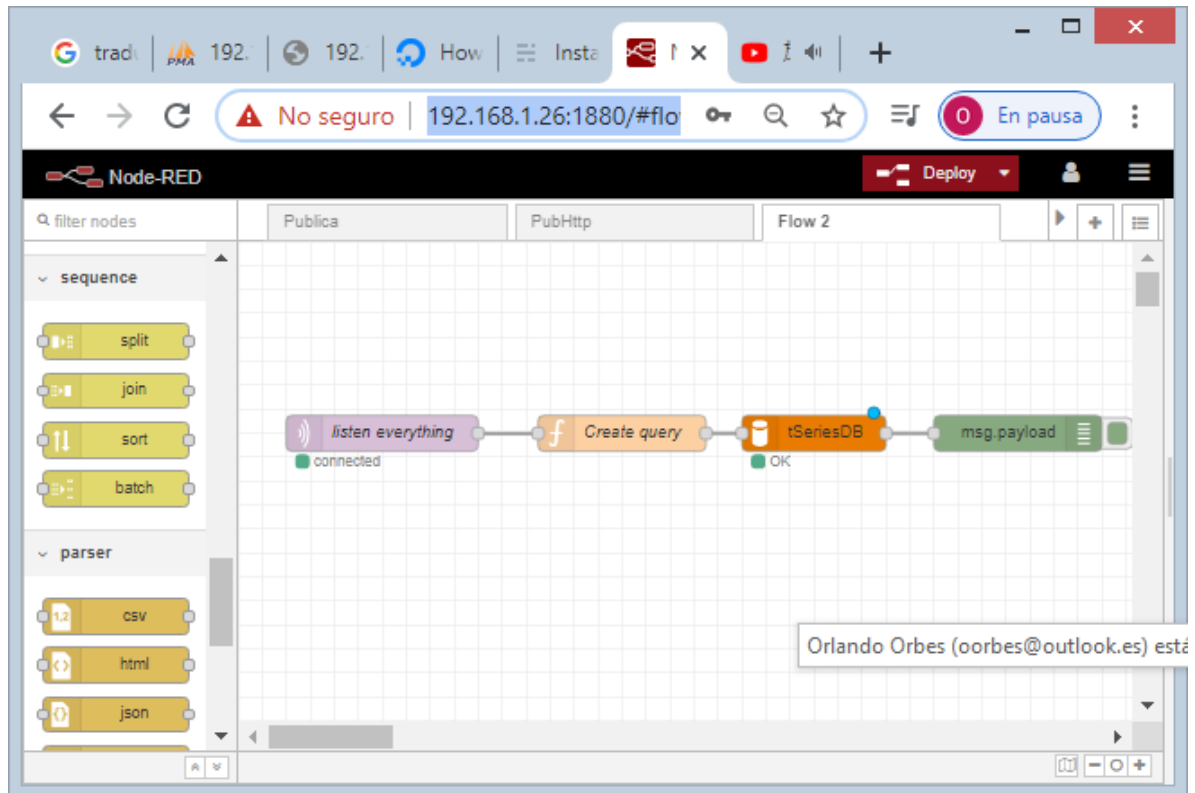


Se edita, configura y despliega el flujo que usa una conexión http para la publicación de información. Se verifica desde consola la efectividad de la orden.

```
orlando@labelect: ~  
orlando@labelect:~$ curl -X POST "192.168.1.26:1880/pub/myTopic/myPayload" -i  
HTTP/1.1 200 OK  
X-Powered-By: Express  
Access-Control-Allow-Origin: *  
X-Content-Type-Options: nosniff  
Content-Type: application/json; charset=utf-8  
Content-Length: 49  
ETag: W/"31-o9Z6ccyOH/ruMT96NeXv4d3hJ9g"  
Date: Mon, 15 Jun 2020 00:11:03 GMT  
Connection: keep-alive  
  
{ "success": true, "message": "published /myPayload" }  
orlando@labelect:~$  
orlando@labelect:~$ curl -X POST "192.168.1.26:1880/pub/myTopic/myPayload" -i  
HTTP/1.1 200 OK  
X-Powered-By: Express  
Access-Control-Allow-Origin: *  
X-Content-Type-Options: nosniff  
Content-Type: application/json; charset=utf-8  
Content-Length: 49  
ETag: W/"31-o9Z6ccyOH/ruMT96NeXv4d3hJ9g"  
Date: Mon, 15 Jun 2020 00:12:13 GMT  
Connection: keep-alive  
  
{ "success": true, "message": "published /myPayload" }  
orlando@labelect:~$
```

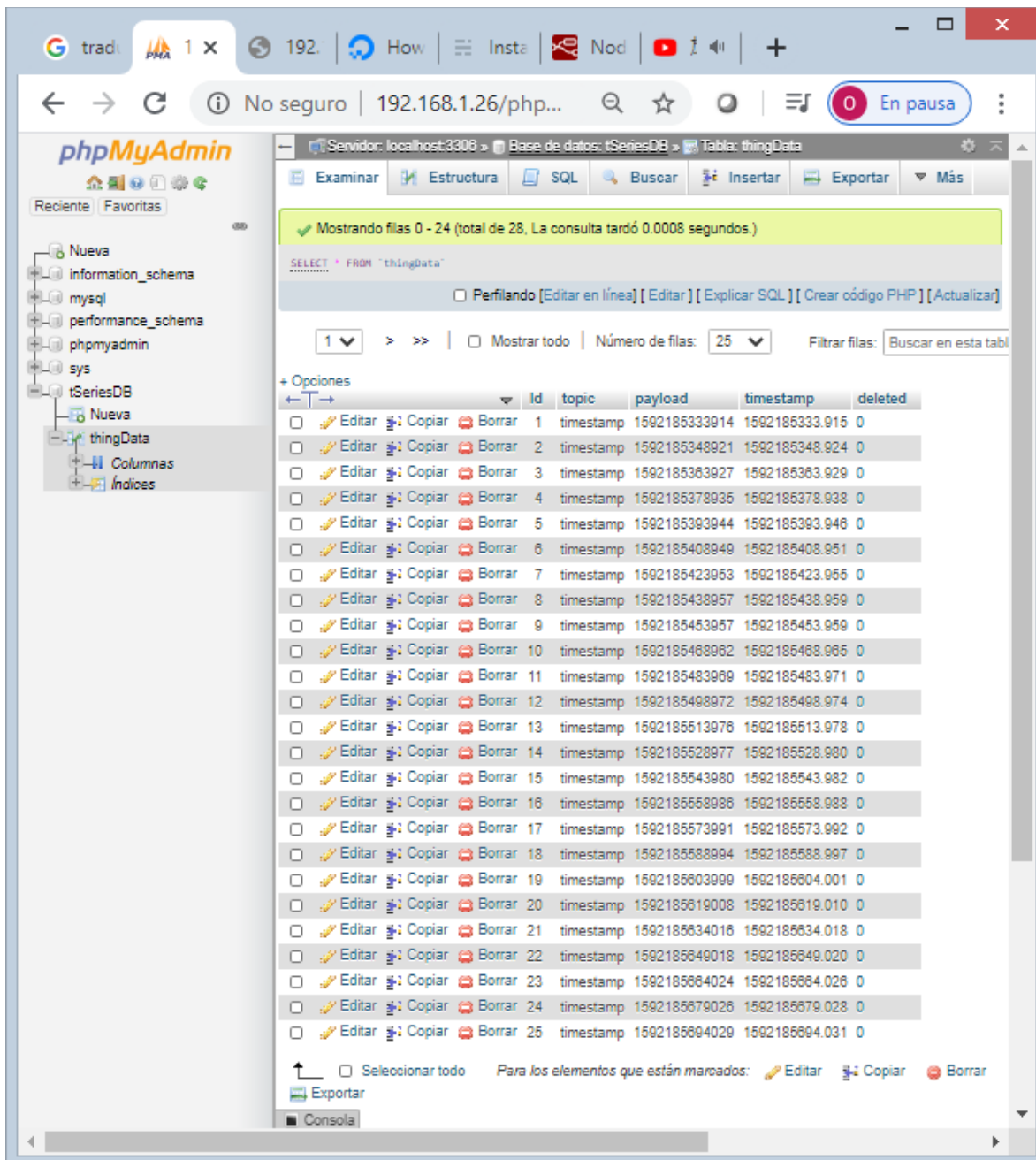
Creación del escuchador de base de datos. Database Listener.

Escucha la corriente de mensajes y almacena en la base de datos. La corriente habilitada está realizada con MQTT, para lo cual se realizará el escuchador.



Realizando la configuración de la base de datos, reportó problemas con la conexión con la cuenta creada con un password corto, se probó con la del root y accedió sin inconvenientes.

La base de datos empezó a registrar mensajes producidos por la recepción de la base de tiempo.



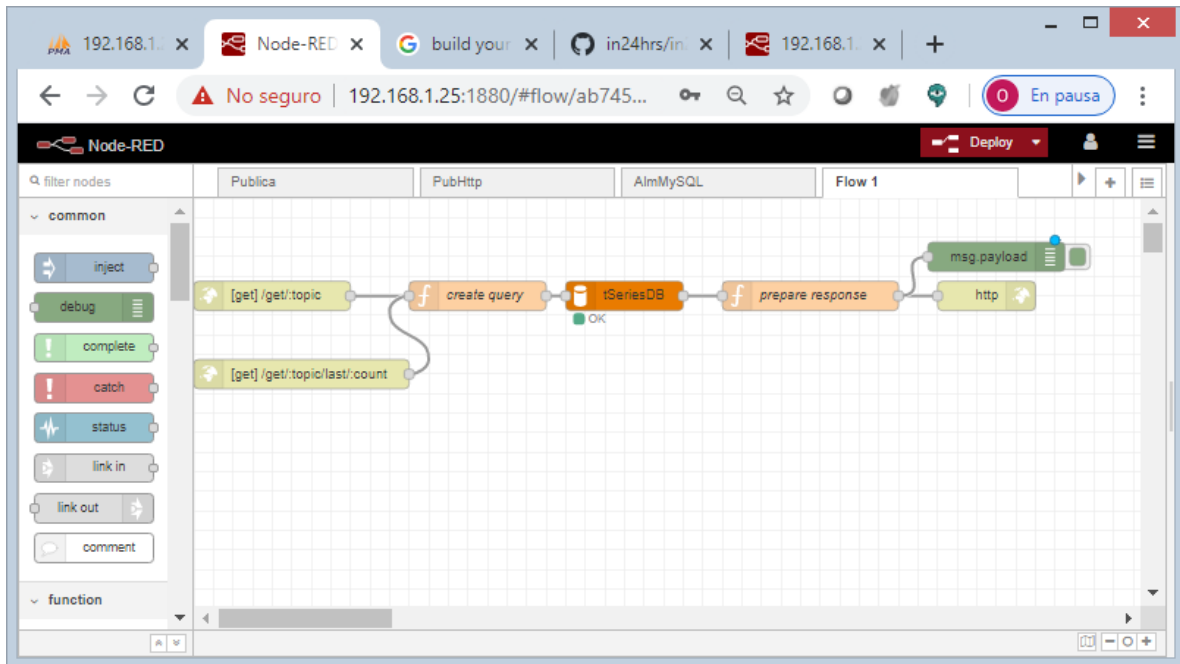
El código generado en node- es el siguiente:

```
{
  "id": "27bc03b9.5d540c",
  "type": "tab",
  "label": "Publica",
  "disabled": false,
  "info": ""
},
{
  "id": "c535c1b4.ec75d",
  "type": "tab",
  "label": "PubHttp",
  "disabled": false,
  "info": ""
},
{
  "id": "3ffa7f7f.fd35a",
  "type": "tab",
  "label": "Flow 2",
  "disabled": false,
  "info": ""
},
{
  "id": "73a6de45.3a216",
  "type": "mqtt-
broker",
  "z": "",
  "name": "",
  "broker": "localhost",
  "port": "1883",
  "clientid": "node-
red",
  "usetls": false,
  "compatmode": true,
  "keepalive": "60",
  "cleansession": true,
  "birthTopic": "",
  "birthQos": "0",
  "birthPayload": "",
  "closeTopic": "",
  "closeQos": "0",
  "closePayload": "",
  "willTopic": "",
  "willQo
```


Usuario: orlando Contraseña ingreso: orl@ndo2020.

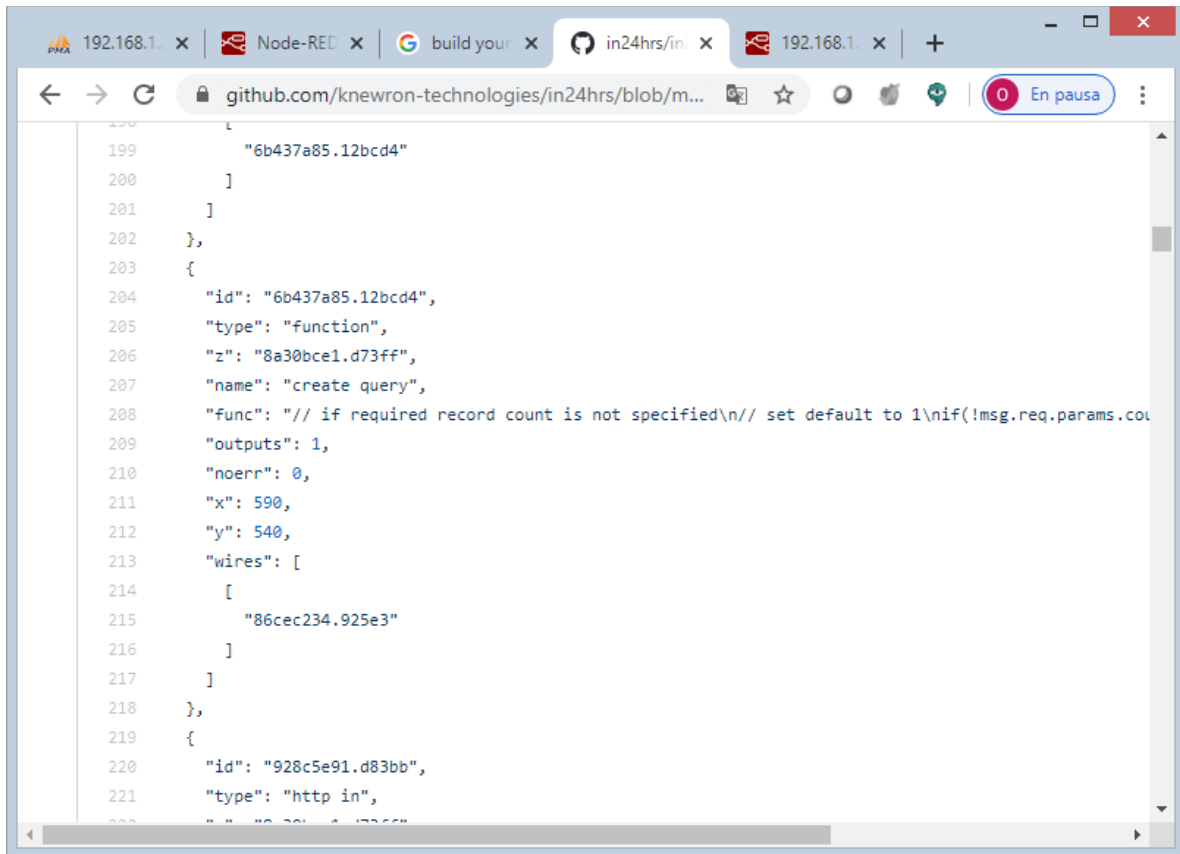
Indicador de mensajes REST API Message Retriever

Se crean dos mensajes para entrega de información almacenada en la base de datos. D1. Obtener un solo registro de datos , D1. Obtener varios registros de datos en una serie.



Se presentaron problemas en la obtención de información, fue preciso activar nuevamente el bróker, a continuación se editó la función de preparar respuesta, la cual estaba vacía pero esto se verificó desde el github del autor, tras examinar el archivo asociado con el programa:

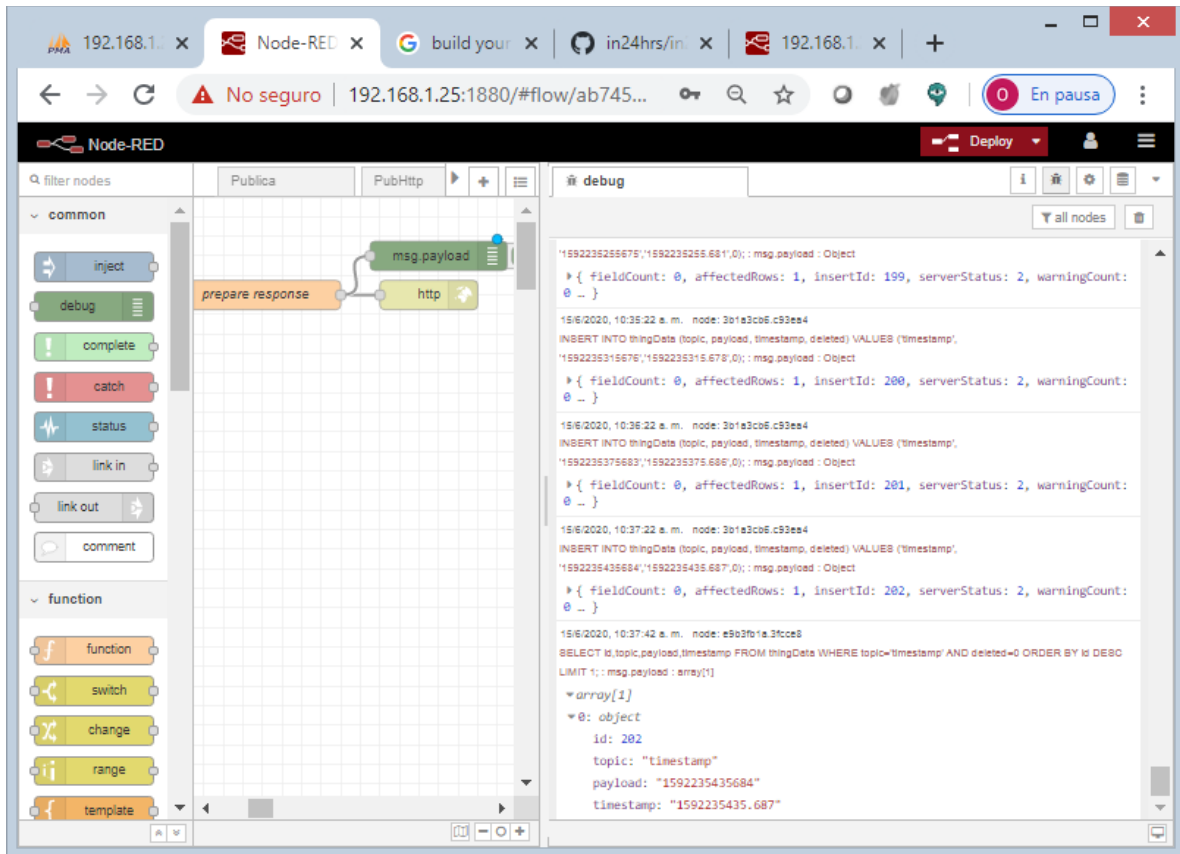
https://github.com/knewron-technologies/in24hrs/blob/master/NodeRED/in24hrs_flow.json



The screenshot shows a web browser window with multiple tabs. The active tab is 'in24hrs/in...' showing a GitHub repository page for 'knewron-technologies/in24hrs/blob/m...'. The main content area displays a JSON configuration for a Node-RED flow, with line numbers 199 through 223 visible on the left. The JSON code is as follows:

```
199     "6b437a85.12bcd4"
200   ]
201 ]
202 },
203 {
204   "id": "6b437a85.12bcd4",
205   "type": "function",
206   "z": "8a30bce1.d73ff",
207   "name": "create query",
208   "func": "// if required record count is not specified\n// set default to 1\nif(!msg.req.params.count){\n  msg.req.params.count = 1;\n}\n",
209   "outputs": 1,
210   "noerr": 0,
211   "x": 590,
212   "y": 540,
213   "wires": [
214     [
215       "86cec234.925e3"
216     ]
217   ]
218 },
219 {
220   "id": "928c5e91.d83bb",
221   "type": "http in",
222   "z": "8a30bce1.d73ff",
223   "x": 100,
224   "y": 600,
225   "wires": [
226     [
227       "928c5e91.d83bb"
228     ]
229   ]
230 }
```

El código ingresado y configuración corresponde al del libro. Sin embargo una mala interpretación del comando desde consola ordenando la obtención de información obligó a examinar en detalle la información de la base de datos y la generación del mensaje que se hizo desde node-red agregando un nodo de depuración.



En la ventana se observa la respuesta correcta, luego de remplazar en la orden de SELECT el nombre apropiado del t3pico: timestamp.

La base de datos trabaja correctamente, s3lo con la cuenta de usuario root, lo que es necesario ajustar debido a que se sugiere el uso de una cuenta con privilegios 3nicamente para el acceso a la base de datos para el proyecto.

EJECUCION Node-RED en el fondo continuamente.

Utilidad de construcci3n usando Node.js llamada forever.

Se instala desde el administrador

npm install forever -g

Después de un reinicio, se ejecutarán los procesos así:

En el PC:

Activar la máquina virtual en VirtualBox: ServUb

Contraseña de ingreso: Orl@ndo2020.

Se ha habilitado el ingreso a través de modo gráfico.

Desde PC, se puede habilitar estación remota así:

Putty

Host Name: orlando@192.168.1.25

Sesion grabada: SSH Tunel

La sesión debe configurarse a través del puerto 5901

Broker:

```
# sudo mosquitto -c /etc/mosquitto/conf.d/broker.conf -v >
mosquitto.log &
```

Base de datos

Desde el explorador:

<http://192.168.1.25/phpmyadmin/>

A continuación usuarios registrados

Usuario root contraseña 24DCKQsbNBzon4sn

Node -red

Desde consola: sudo node-red

Desde explorador:

<http://192.168.1.25:1880>

Usuario admin

Contraseña Nod20red+

Para Node -red se prueba la opción de autorespaldo

```
# forever start -l node-red.log --append /usr/local/bin/node-red
```

Pruebas de conexión

```
# curl -X GET "192.168.1.25:1880/get/timestamp/last/3"
```

```
# curl -X GET "192.168.1.25:1880/get/timestamp"
```

Se modifica el archivo broker.conf para admitir depuración mediante registro silencioso de las actividades como conexiones, suscripciones, publicaciones, desconexiones, errores y advertencias. Los detalles son almacenados cada 15 minutos. Se habilita también persistencia en el archivo broker_log.txt. Las conexiones anónimas son detenidas y se habilita el archivo que contiene los usuarios y contraseñas para autenticación. Recomendación al agregar usuarios y sus contraseñas debe ajustarse los privilegios del archivo pwss.

Usuarios existentes: labelect contraseña labelect, Finalmente se lanza el servicio:

```
#mosquitto -c /etc/mosquitto/conf.d/broker.conf -v > mosquitto.log &
```

Desarrollo de microservicios

Una vez se prueba la efectividad de los flujos en nodered asociados con:

- Habilitación de la publicación del timestamp a través de mosquitto para todos los suscriptores de topic.
- Publicación de mensajes desde http usando el broker de mosquitto mediante el uso de pub y curl, así como de get y de orden desde explorador,
- Puesta en marcha el listener para escucha de todos los mensajes en mqtt y su almacenamiento en mysql
- Atención de solicitudes de acceso a los registros de base de datos ya sea último o de un determinado número de registros.

Se procede a configurar la activación de node-red de forma automática. Se reinició el sistema y se confirmó que todo se encuentra en orden, no hay mensajes de error ni advertencias en el inicio del servidor.

Instalación de Workbench Mysql

Esta herramienta permite el modelado y descripción de las bases de datos, se ha instalado y asiste al curso en youtube.

Pautas instalación:

<https://platzi.com/tutoriales/1566-bd/8226-como-instalar-mysql-y-workbench-en-ubuntu-sin-morir-en-el-intento/>

Curso:

Aumentar espacio de disco asignado

La capacidad por defecto de 14Gb está cerca de ser alcanzada por lo que es necesario realizar una ampliación de la capacidad asignada, sabiendo que el disco duro inicial es de 30Gb. Se amplía también la capacidad de disco desde VMW de 30Gb a 60Gb, pero no se ajusta la partición.

Los comandos usados se obtienen del vídeo:

<https://www.youtube.com/watch?v=nkJvqfYmyLU>

min 9:05.

Son los siguientes:

Informe de uso de las particiones:

`sudo df -h`

Se identifica las particiones con limitación de espacio libre.

`sudo pvs`

Informa sobre los grupos de volúmenes

`vgdisplay`

Da detalles de uso y espacio disponible.

`fdisk -l`

Etiqueta del disco

`lvextend -L +10G /dev/mapper/ubuntu--vg-ubuntu--lv`

Extiende en 10G la unidad que tiene limitación

`resize2fs -p /dev/mapper/ubuntu--vg-ubuntu--lv`

Hace efectivo el ajuste

