

Trabajo Practico Final



Materia: Laboratorio de Sistemas Operativos y Redes

Profesor: José Luis Di Biase

Integrantes: Cavallin, Matías
Lamandia, Ignacio
Veliez, Gastón

Indice

1.Introducción	3
2.Instalación	3
3.Configuración	4
4.Ejecución	6
5.Problemas	7
6.Fuentes	8

1.Introducción

¿Que es pump.io?

Pump es una red social general. Además de la evidente ventaja de ser descentralizada y libre, en Pump puedes publicar texto con formato, imágenes o lo que quieras (sus posibilidades aún están por explotar). No hay limitación de caracteres ni de ninguna otra clase.

Descripción del proyecto

En este documento vamos a explicar paso a paso la Instalación, configuración, y dar una demostración de la ejecución de la red social open source pump.io.

2.Instalación

Software adicional necesario:

1. node.js 0.8.0 o superior
2. npm 1.1.0 o superior
3. Un servidor de base de datos (Redis, MongoDB , etc)
4. GraphicsMagick
5. ImageMagick

Pasos a seguir:

Instalar paquetes esenciales

```
sudo apt-get install build-essential curl git python-software-properties
```

Instalar NodeJS

```
sudo add-apt-repository ppa:chris-lea/node.js
```

```
sudo apt-get update  
sudo apt-get install nodejs
```

Instalar dependencias de sistema especificas de pump.io

```
sudo apt-get install redis-server graphicsmagick screen
```

Instalar servidor de base de datos (En nuestro caso Redis)

```
sudo apt-get install redis-server
```

Instalación de Pump.io

Clonamos el repositorio

```
git clone https://github.com/e14n/pump.io.git
```

Instalamos las dependencias usando npm:

```
cd pump.io  
sudo npm install
```

Instalamos el databank de la base de datos elegida, en nuestro caso redis.

```
cd node_modules/databank  
sudo npm install databank-redis
```

3.Configuración

Configuración del sistema:

1- Creamos el directorio para las subidas de los usuarios:

```
mkdir pump.io/uploads
```

2- Movemos el directorio donde tenemos pump.io al directorio srv:

```
mv ~/pump.io /srv/
```

Configuración de Pump.io:

1- Copiamos el archivo de configuración por defecto al directorio /etc :

```
cp /srv/pump.io/pump.io.json.sample /etc/pump.io.json
```

2- Editamos el archivo de configuración acorde a nuestras preferencias:

```
nano /etc/pump.io.json
```

Configuración utilizada:

```
{
```

```
  "driver": "redis", ## Base de datos a utilizar.
```

```
  "params": {}, ## Parametros para la base de datos.
```

```
  "secret": "somethingverysecrethere", ## Secreto.
```

```
  "noweb": false, ## Habilitacion o no de la interfaz web (Seteado en false para tener interfaz web).
```

```
  "site": "proyecto.io", ## Nombre del sitio.
```

```
  "owner": "Gaston Veliez, Ignacio Lamandia y Matias Cavallin",  
## Propietarios del sitio.
```

```
  "ownerURL": "", ## Link del/los propietarios.
```

```
  "hostname": "proyecto.io", ## Nombre del host.
```

```
  "address": "127.0.0.1", ## Direccion del host.
```

```
  "port": 80, ##Puerto utilizado.
```

```
  "nologger": false, ## Habilitacion o no del loggeo (Seteado en false para habilitarlo).
```

```
  "serverUser": "pumpio", ## Usuario de pumpio.
```

```
  "uploaddir": "/srv/pump.io/uploads", ## Directorio donde se almacenan los archivos subidos.
```

"debugClient": false, ## Seteado en false para la mayoría de los usuarios. Esta opción es mas que nada para desarrolladores.

"firehose": "ofirehose.com", ## firehose se encarga de distribuir las notas publicadas a la red publica (por default ofirehose.com).

"disableRegistration": false, ## Deshabilitacion del registro al servidor.

"canUpload": true, ## Permisos para subir archivos.

"sockjs": true, ## Utilizacion de sockjs para conexión en tiempo real.

"requireEmail": false, ## Requerimiento de mail al registrarse.

"noCDN":true ## Utilizar copias de las librerias JavaScript en lugar de las de CDN.

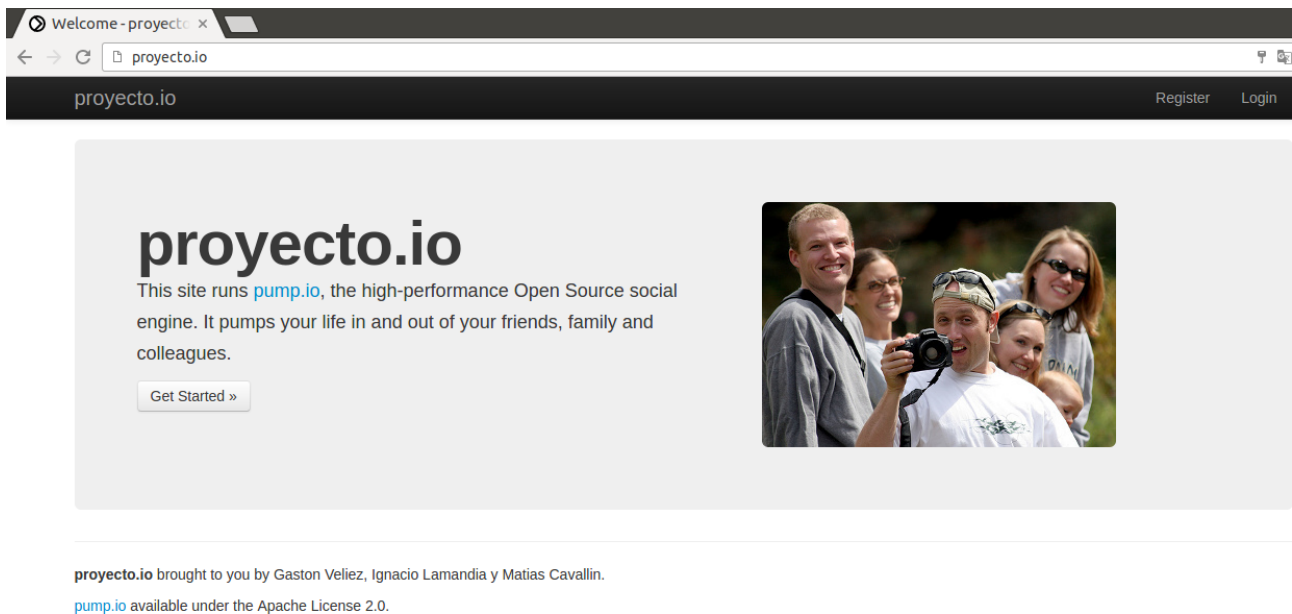
}

4.Ejecución

Para ejecutar vamos al directorio /srv/pump.io y ejecutamos el comando:

```
sudo npm start  
0  
sudo pump
```

Una vez funcionando ingresamos al url de nuestra red social, en este caso proyecto.io



5.Problemas:

-Apache web server:

Tener instalado Apache complica la configuración de pump.io, aunque se pueden configurar para funcionar ambos al mismo tiempo para esta demostración se decidió tener solo instalado pump.io para que sea mas sencilla la instalación y demostración.

-Problemas al publicar nota:

Una vez corriendo pump.io un usuario no podía publicar una nota ya que se conectaba a la web a través de la ip y no de la url.

Solución:

1- Editar el archivo de configuración de hosts :

```
sudo nano /etc/hosts
```

y escribir la linea "127.0.0.1 proyecto.io", 127.0.0.1 siendo la ip que seteamos nosotros en la configuración.

Ademas así en vez de escribir la ip para entrar ahora se hace por proyecto.io.

-Problemas al postear una imagen:

Cuando un usuario quería postear una imagen o deseaba cambiar su avatar no podía y devolvía el error "xhr returned response code 0".

Solución:

1- Ir a la carpeta srv/pump.io

2- Instalar el imagemagick:

```
sudo apt-get install imagemagick
```

```
sudo npm install imagemagick
```

3- Instalar GraphicsMagick a través del comando gm

```
sudo npm install gm
```

4- Darle permisos de lectura y escritura a la carpeta /srv/pump.io/uploads.

-No poder entrar desde otra computadora:

Otro problema que tenemos actualmente es que si una persona quisiera entrar al perfil de otro usuario no puede y si trata agregando http hasta lo lleva a otra pagina que no tiene ninguna relación con el trabajo.

6.Fuentes:

pump.io

github.com/e14n/pump.io/wiki

github.com/e14n/pump.io/issues