

Santiago de Cali, 27 de noviembre de 2015

William Martín Chávez González-11203008

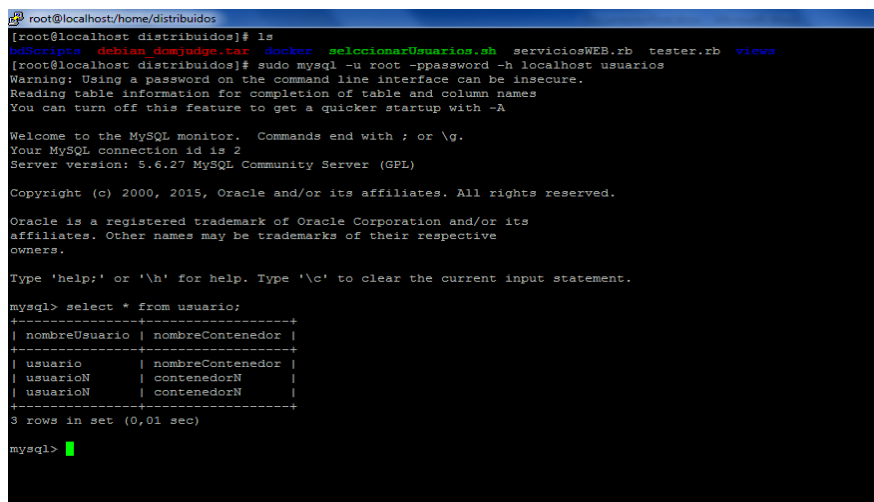
PROYECTO FINAL SISTEMAS OPERATIVOS DISTRIBUIDOS

En primera instancia, se tiene la configuración de Docker en la MaqServidor_CentOS7 con la imagen creada en debían del DOMJudge. Ésta nos permitirá crear los contenedores una vez le demos run. Debemos hacerlo en background para tener la posibilidad de ejecutar otros procedimientos al tiempo.



```
root@localhost:/home/distribuidos
login as: root
root@192.168.1.50's password:
Last login: Sun Nov 29 19:33:54 2015
[root@localhost ~]# cd /home/distribuidos/
[root@localhost distribuidos]# ls
debian_domjudge.tar  docker  serviciosWEB.rb  tester.rb  views
[root@localhost distribuidos]# docker ps -l
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
[root@localhost distribuidos]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             VIRTUAL SIZE
debian_domjudge     latest             153bade60b69       5 days ago         97.6 MB
```

Igualmente, se debe configurar la base de datos Mysql y correrla en backgroud para que esté recibiendo solicitudes de CRUD sin perder la posibilidad de ejecutar otras operaciones en paralelo.



```
root@localhost:/home/distribuidos
[root@localhost distribuidos]# ls
debian_domjudge.tar  docker  seleccionarUsuarios.sh  serviciosWEB.rb  tester.rb  views
[root@localhost distribuidos]# sudo mysql -u root -ppassword -h localhost usuarios
Warning: Using a password on the command line interface can be insecure.
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.6.27 MySQL Community Server (GPL)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> select * from usuario;
+-----+-----+
| nombreUsuario | nombreContenedor |
+-----+-----+
| usuario      | contenedorN      |
| usuarioN     | contenedorN      |
| usuarioN     | contenedorN      |
+-----+-----+
3 rows in set (0,01 sec)

mysql>
```

Se tienen 3 usuarios de prueba creados.

ARQUITECTURA DEL SISTEMA: (Para una mejor vista ver archivo de imagen adjunta).



- **Nodos de procesamiento:**

1. **Usuario:** Son los usuarios que acceden a los servicios WEB que ofrece el servidor.
2. **Servidor:** Encargado de ofrecer los servicios web. Tiene Docker instalado, una base de datos MySQL y un módulo Administrador en Ruby para el manejo de solicitudes y peticiones WEB con Sinatra.
3. **Debian_DOMJudgeN:** Son los N contenedores que se levantan con las n peticiones o conexiones de Usuarios que acceden al servidor.

Cuando un cliente o **usuarioN** desea ejecutar los servicios del **DomJuge**, primero debe pasar por el filtro del Servidor con el administrado de aplicaciones hecho en Ruby. Para tal fin, el administrador maneja 3 archivos(ver documentos adjuntos): **tester.rb**, la lógica, o clase Control, que ejecuta las funciones necesarias para listar, registrar o eliminar usuarios por pedido, además tiene un método(<http://stackoverflow.com/questions/5029427/ruby-get-local-ip-nix>) para calcular la dirección IP del servidor y asignársela a la respuesta del cliente en su petición de domjudge; **serviciosWEB.rb** que se encarga de ofrecer los servicios en Sinatra, get y post, de las interfaces que

despliega el componente Administrador; e **interfaz.erb** que despliega un formulario para la ejecución por vista de los servicios.

A continuación se prueban los siguientes comandos:

- Primeramente no hay ningún contenedor en Docker activado como se muestra:

```
root@localhost/home/distribuidos
[root@localhost distribuidos]# ls
bdScripts  debian  domjudge.tar  docker  seleccionarUsuarios.sh  serviciosWEB.rb  tester.rb  views
[root@localhost distribuidos]# docker ps -l
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS              NAMES
[root@localhost distribuidos]#
```

- Una vez se corre el servicio web, La interfaz desplegada en la aplicación es ésta. Muestra la IP del servidor que está corriendo en éste momento(se le puede verificar por consola).

```
[root@localhost distribuidos]# ruby serviciosWEB.rb -es
[2015-11-30 20:01:56] INFO  WEBrick 1.3.1
[2015-11-30 20:01:56] INFO  ruby 2.0.0 (2014-11-13) [x86_64-linux]
== Sinatra (v1.4.6) has taken the stage on 4567 for s with backup from WEBrick
[2015-11-30 20:01:56] INFO  WEBrick::HTTPServer#start: pid=3857 port=4567
```



```
[root@localhost distribuidos]# ruby serviciosWEB.rb -es
[2015-11-30 20:01:56] INFO WEBrick 1.3.1
[2015-11-30 20:01:56] INFO ruby 2.0.0 (2014-11-13) [x86_64-linux]
== Sinatra (v1.4.6) has taken the stage on 4567 for s with backup from WEBrick
[2015-11-30 20:01:56] INFO WEBrick::HTTPServer#start: pid=3857 port=4567
Se ha iniciado el controlador del servidor,IPServidor:10.0.2.15
WARN: tilt autoloading 'tilt/erb' in a non thread-safe way; explicit require 'tilt/erb' suggested.
172.30.161.19 - - [30/Nov/2015:20:02:37 +0100] "GET / HTTP/1.1" 200 648 0.0065
172.30.161.19 - - [30/Nov/2015:20:02:37 CET] "GET / HTTP/1.1" 200 648
- -> /
```

- Una vez activado el servicio se pueden crear los usuarios. Puede verificarlo en consola tanto la creación del contenedor como la creación del usuario en MySQL. Todo se muestra a continuación.



```
[root@localhost distribuidos]# ruby serviciosWEB.rb -es
[2015-11-30 20:45:39] INFO WEBrick 1.3.1
[2015-11-30 20:45:39] INFO ruby 2.0.0 (2014-11-13) [x86_64-linux]
== Sinatra (v1.4.6) has taken the stage on 4567 for s with backup from WEBrick
[2015-11-30 20:45:40] INFO WEBrick::HTTPServer#start: pid=4707 port=4567
Se ha iniciado el controlador del servidor,IPServidor:10.0.2.15
WARN: tilt autoloading 'tilt/erb' in a non thread-safe way; explicit require 'tilt/erb' suggested.
172.30.161.19 - - [30/Nov/2015:20:45:45 +0100] "GET / HTTP/1.1" 200 648 0.0067
172.30.161.19 - - [30/Nov/2015:20:45:45 CET] "GET / HTTP/1.1" 200 648
- -> /
Se ha iniciado el controlador del servidor,IPServidor:10.0.2.15
Se le ha asociado el domjudge:domjudge1, con direccion IP:10.0.2.15, por el puerto:5000
sh: -c: línea 0: error sintáctico cerca del elemento inesperado `('
sh: -c: línea 0: `mysql @sql_args insert into usuario(nombreUsuario,nombreContenedor) values('usuario1','domjudge1');'
172.30.161.19 - - [30/Nov/2015:20:45:56 +0100] "POST /registrarse HTTP/1.1" 200 673 0.0104
172.30.161.19 - - [30/Nov/2015:20:45:56 CET] "POST /registrarse HTTP/1.1" 200 673
http://172.30.161.172:4567/ -> /registrarse
08ca5c5178338c65a5892ebe88ea35309899175722f284c1f8540322fe8dae4b
```


Domjudge en background:

```
[root@localhost ~]# docker ps -l
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
08ca5c517833   debian_domjudge  "/bin/bash"            3 minutes ago  Up 3 minutes  0.0.0.0:5000->80/tcp
dge1
[root@localhost ~]#
```

Verificación de acceso al contenedor mediante URL- En este caso la IP arroja 10.0.2.15, pero en la interfaz puente es la indicada en el servidor:

```
[root@localhost ~]# docker ps -l
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
08ca5c517833   debian_domjudge  "/bin/bash"            3 minutes ago  Up 3 minutes  0.0.0.0:5000->80/tcp
dge1
[root@localhost ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 08:00:27:ff:2b:09 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 78079sec preferred_lft 78079sec
    inet6 fe80::a00:27ff:feff:2b09/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 08:00:27:99:e5:54 brd ff:ff:ff:ff:ff:ff
    inet 172.30.161.172/16 brd 172.30.255.255 scope global dynamic enp0s8
        valid_lft 11774sec preferred_lft 11774sec
    inet6 fe80::a00:27ff:fe99:e554/64 scope link
        valid_lft forever preferred_lft forever
```

← → ↻ 172.30.161.172:5000

 **Apache2 Debian Default Page**

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented** in `/usr/share/doc/apache2/README.Debian.gz`. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/
|-- apache2.conf
/   |-- ports.conf
|-- mods-enabled
/   |-- *.load
/   |-- *.conf
|-- conf-enabled
/   |-- *.conf
|-- sites-enabled
/   |-- *.conf
```

- Lamentablemente no se puede observar lo que arroja el IO.pope- de manera coherente, tal vez por formato- que permite ver la salida en consola de los usuarios registrados en la base de datos.



- Para eliminar los usuarios no está disponible la funcionalidad correctamente por un problema de reconocimiento de variable JSON. Sin embargo las funciones y la vista permiten ejecutarlo.

A continuación se presentan las dificultades presentadas:

Solución https://www.domjudge.org/docs/admin-manual-3.html#install_config:package-install

```
domjudge@a23de14ab9be:~/domjudge-5.0.1$ make domserver && sudo make install-domserver
bash: make: command not found
domjudge@a23de14ab9be:~/domjudge-5.0.1$ ./configure
checking whether configure should try to set CFLAGS... yes
checking whether configure should try to set CXXFLAGS... yes
checking whether configure should try to set LDFLAGS... yes
checking for gcc... no
checking for cc... no
checking for cl.exe... no
configure: error: in `/home/domjudge/domjudge-5.0.1':
configure: error: no acceptable C compiler found in $PATH
See `config.log' for more details
domjudge@a23de14ab9be:~/domjudge-5.0.1$ ls
```

No levanta el servicio de evaluación:

```
root@a23de14ab9be:~/domjudge/judgehost/bin# service mysql start
[ ok ] Starting MySQL database server: mysqld already running.
root@a23de14ab9be:~/domjudge/judgehost/bin# service apache2 start
[ ok ] Starting web server: apache2.
root@a23de14ab9be:~/domjudge/judgehost/bin# ./judgedaemon
[Nov 28 12:58:45] judgedaemon[16862]: Judge started on a23de14ab9be [DOMjudge/5.0.1]
judgedaemon[16862]: Judge started on a23de14ab9be [DOMjudge/5.0.1]

judgedaemon[16862]: Installing signal handlers

[Nov 28 12:58:45] judgedaemon[16862]: error: Cannot read REST API credentials file /root/domjudge/judgehost/etc/restapi.secret
judgedaemon[16862]: error: Cannot read REST API credentials file /root/domjudge/judgehost/etc/restapi.secret

root@a23de14ab9be:~/domjudge/judgehost/bin#
```

- **Solución:** https://www.domjudge.org/snapshot/admin-manual/admin-manual-3.html#install_config:configurablescripts: The judgehost connects to the domserver via a REST API. You need to create an account for the judgedaemons to use (this may be a shared account between all judgedaemons) with a difficult, random password and the 'judgehost' role. On each judgedaemon, create a file etc/restapi.secret containing the id, URL, username and password whitespace-separated on one line, for example:
 - default `http://example.edu/domjudge/api/ judgehosts`
`MzfJYWF5agSIUfmiGEy5mgkfqU`
 - Multiple lines may be specified to allow a judgedaemon to work for multiple domservers. The id is used to differentiate between multiple domservers, and should be unique within therestapi.secret file.
- Then start the judgedaemon: `bin/judgedaemon`

- **Problema derivado:** Permiso denegado para acceder al servidor.

```
root@a23de14ab9be:~/domjudge/judgehost/bin# ./judgedaemon
[Nov 28 13:43:15] judgedaemon[17108]: Judge started on a23de14ab9be [DOMjudge/5.0.1]
judgedaemon[17108]: Judge started on a23de14ab9be [DOMjudge/5.0.1]

judgedaemon[17108]: Installing signal handlers

[Nov 28 13:43:15] judgedaemon[17108]: Chroot disabled. This reduces judgehost security.
judgedaemon[17108]: Chroot disabled. This reduces judgehost security.

[Nov 28 13:43:15] judgedaemon[17108]: Not using cgroups. Using cgroups is highly recommended. See the manual for details.
judgedaemon[17108]: Not using cgroups. Using cgroups is highly recommended. See the manual for details.

[Nov 28 13:43:15] judgedaemon[17108]: Registering judgehost on endpoint http://localhost/domjudge/api
judgedaemon[17108]: Registering judgehost on endpoint http://localhost/domjudge/api

judgedaemon[17108]: API request POST judgehosts

[Nov 28 13:43:15] judgedaemon[17108]: error: Error while executing curl POST to url http://localhost/domjudge/api/judgehosts: http status code: 403, response: <!DOCTYPE
HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>403 Forbidden</title>
</head><body>
<h1>Forbidden</h1>
<p>You don't have permission to access /domjudge/api/judgehosts
on this server.<br />
</p>
<hr>
<address><code>Apache/2.4.10 (Debian) Server at localhost Port 80</code></address>
</body></html>

judgedaemon[17108]: error: Error while executing curl POST to url http://localhost/domjudge/api/judgehosts: http status code: 403, response: <!DOCTYPE HTML PUBLIC "-//I
ETF//DTD HTML 2.0//EN">
<html><head>
<title>403 Forbidden</title>
</head><body>
<h1>Forbidden</h1>
<p>You don't have permission to access /domjudge/api/judgehosts
on this server.<br />
</p>
<hr>
<address><code>Apache/2.4.10 (Debian) Server at localhost Port 80</code></address>
</body></html>
```

REFERENCIAS:

- <http://stackoverflow.com/questions/5029427/ruby-get-local-ip-nix>
- <https://www.domjudge.org/snapshot/admin-manual/admin-manual-3.html>
- <https://www.domjudge.org/docs/admin-manual-3.htm>
- <https://devops.profitbricks.com/tutorials/install-mysql-on-centos-7/>
- <http://www.tutorialspoint.com/mysql/mysql-insert-query.htm>