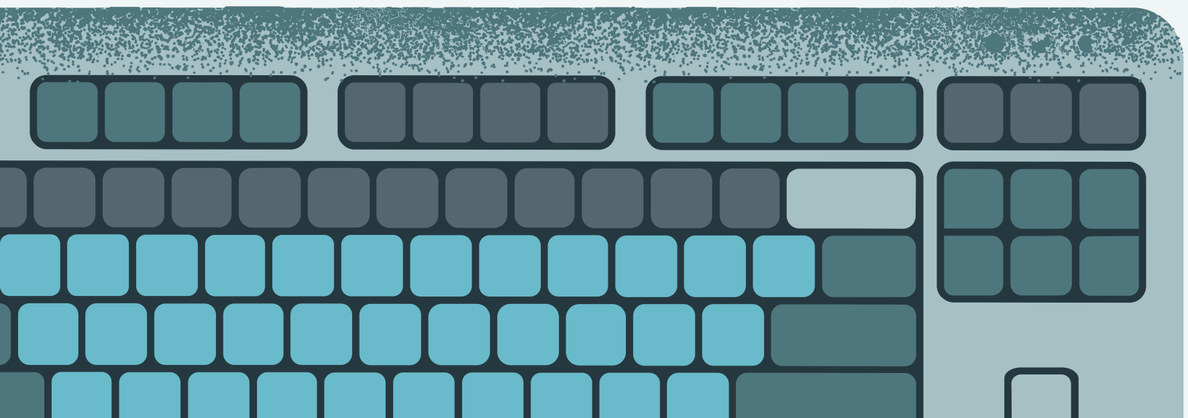
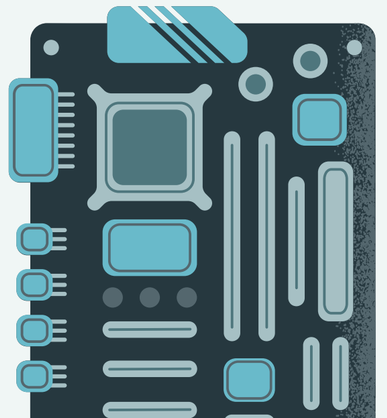
Proyecto Realizado por Miguel Sotelo - William Banguera

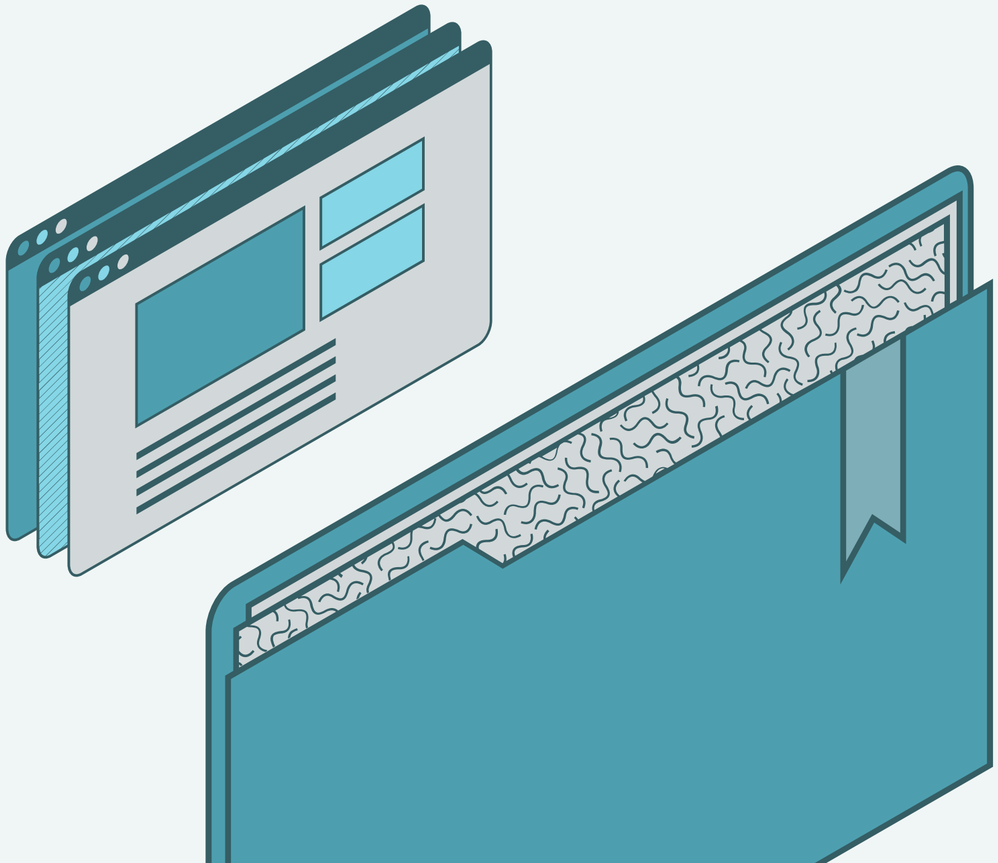
PROYECTO SERVICIOS TELEMÁTICOS

BALANCEO DE CARGA DE BASES DE DATOS CON MYSQL Y HAPROXY



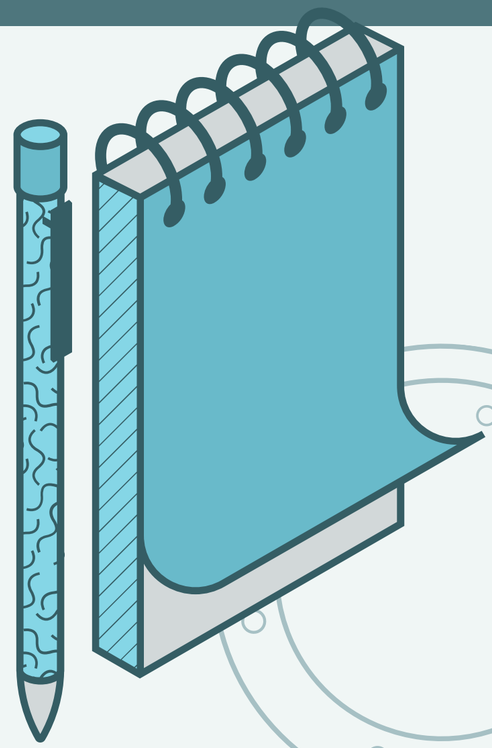



ÍNDICE



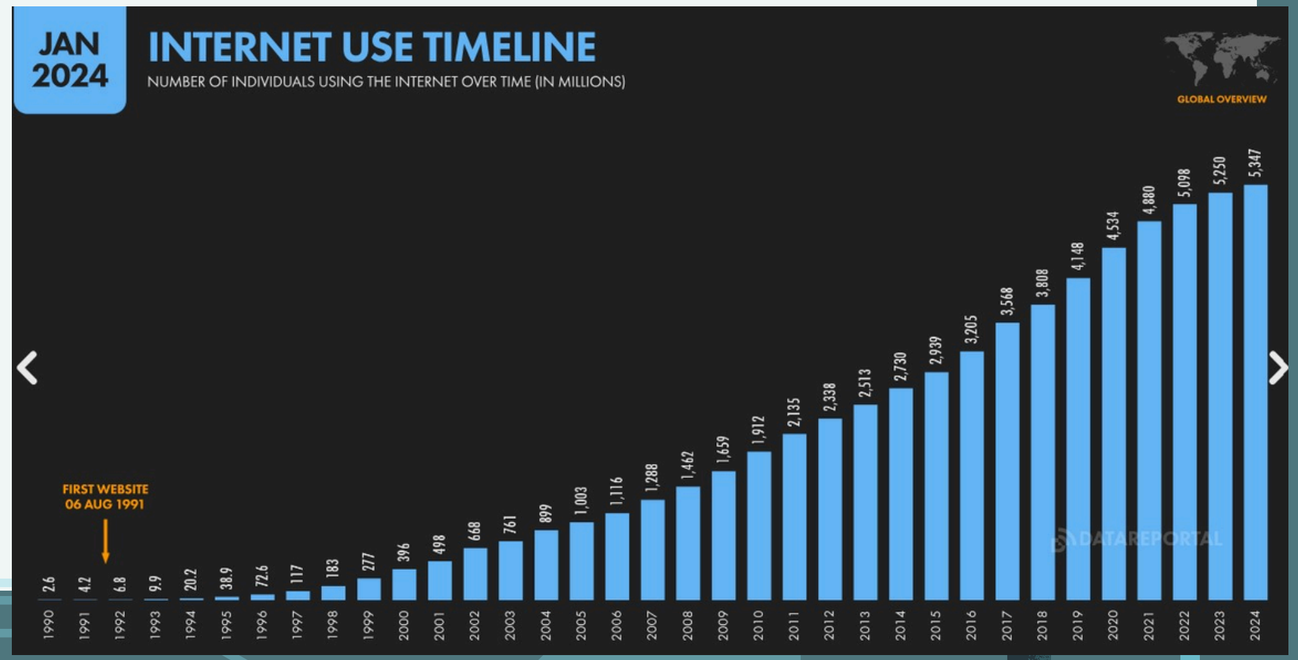
INTRODUCCIÓN

Este proyecto presenta una solución de balanceo de carga para bases de datos MySQL mediante HAProxy, configurada en un entorno virtualizado con Vagrant y VirtualBox. El objetivo es distribuir el tráfico de consultas entre múltiples instancias de MySQL para mejorar la disponibilidad y reducir el tiempo de respuesta. Con esta implementación, se mitigan los riesgos de sobrecarga en un solo servidor, aumentando la eficiencia y estabilidad del servicio en entornos de alta demanda.



PROBLEMA

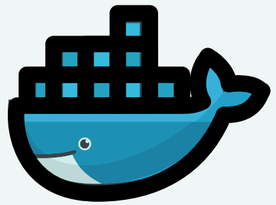
El crecimiento acelerado de usuarios y datos en aplicaciones en línea exige que las bases de datos manejen múltiples solicitudes simultáneas sin comprometer el rendimiento. Sin un mecanismo adecuado, una sola base de datos puede experimentar sobrecarga y fallas, afectando la experiencia del usuario. Este proyecto aborda el problema mediante la implementación de un entorno de balanceo de carga en MySQL utilizando HAProxy, distribuyendo eficientemente el tráfico entre múltiples servidores para asegurar disponibilidad y rendimiento óptimos.

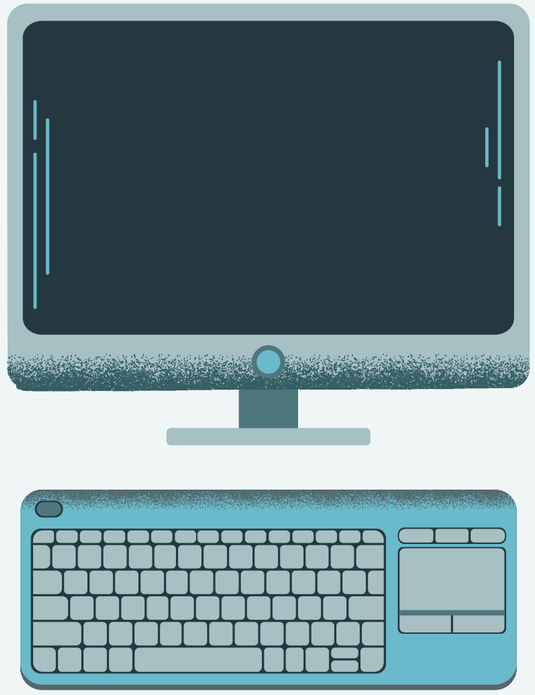


ALTERNATIVAS DE SOLUCIÓN

DOCKER CON CONTENEDORES

- Despliegue y gestión de aplicaciones en contenedores
- Aislamiento a nivel de aplicación (contenedores)
- Ligero y rápido debido al uso compartido de recursos del kernel
- Limitada a configuraciones de red y puertos

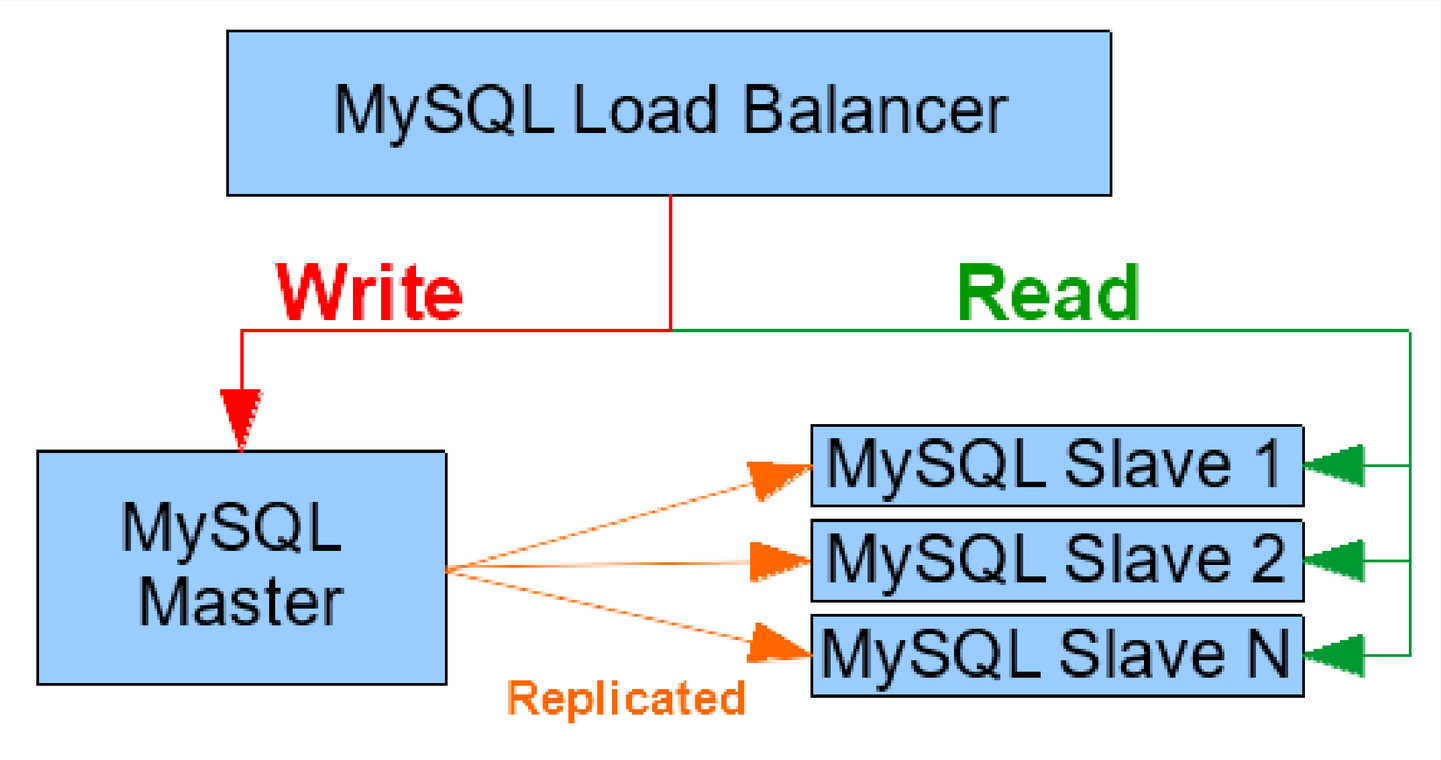




VAGRANT Y UBUNTU

- Entornos de desarrollo reproducibles y configurables
- Crea entornos de desarrollo completos en VMs
- Puede ser más pesado que contenedores, pero flexible
- permite replicar entornos complejos de red y hardware

DISEÑO DE SOLUCIÓN



Entorno: Máquinas virtuales con Vagrant y VirtualBox.

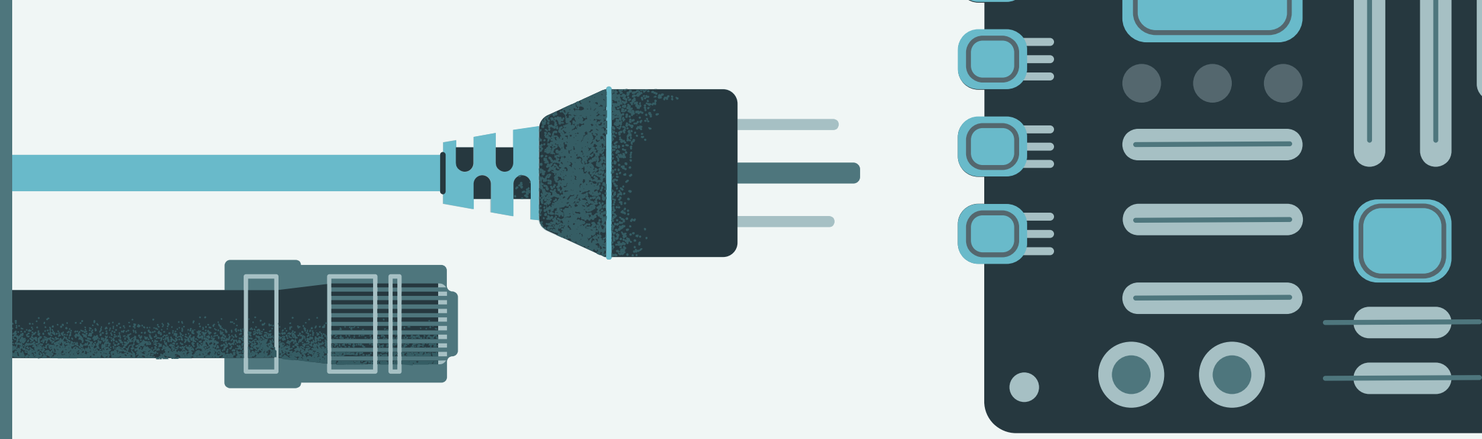
- Configuración:

HAProxy: Balanceador que distribuye solicitudes con algoritmo roundrobin.

- IP privada: 192.168.56.101

Servidores MySQL: Tres máquinas con MySQL para gestionar las consultas.

- db1 (Maestro): 192.168.56.102
- db2 (Esclavo): 192.168.56.103
- db3 (Esclavo): 192.168.56.104



IMPLEMENTACION

En el servidor maestro y en los eslcavos
instalar mysql-server

```
vagrant@db1:~$ sudo apt-get install mysql-server
```

En el archivo my.conf se especifica el
server-id para el maestro y se
descomenta las siguentes lineas

/ETC/MYSQL/MY.CNF

```
#bind-address           = 127.0.0.1

server-id = 1
log_bin = /var/log/mysql/mysql-bin.log
binlog-ignore-db = "mysql"
```

Configuración del esclavo db1

```
#bind-address            = 127.0.0.1

server-id = 2
relay-log = /var/log/mysql/mysql-relay-bin.log
binlog-format = mixed
log_bin = /var/log/mysql/mysql-bin.log
binlog-ignore-db = mysql
```

Configuración del esclavo db2

```
#bind-address            = 127.0.0.1

server-id = 3
relay-log = /var/log/mysql/mysql-relay-bin.log
binlog-format = mixed
log_bin = /var/log/mysql/mysql-bin.log
binlog-ignore-db = mysql
```

Configuración del maestro (DB1)

mysql –u root –p
create user 'replica'@'192.168.56.103' identified by 'replica';
grant replication slave on *.* to 'replica'@'192.168.56.103' identified by 'replica;flush privileges;

Se realiza en ambos los esclavos para crear el usuario para la replicación, esto aplica para la ip 103 y 104

flush table with read lock;
show master status;

Configuración del maestro (DB1)

```
mysql> show master status;
+------------------+----------+--------------+------------------+
| File             | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+------------------+----------+--------------+------------------+
| mysql-bin.000005 |      107 |              | mysql            |
+------------------+----------+--------------+------------------+
1 row in set (0.00 sec)
```

En este paso es importante recordar el nombre File y la position, ya que nos servirá en configuración de esclavos

```
CHANGE MASTER TO MASTER_HOST='192.168.56.102', MASTER_USER='REPLICA',
MASTER_PASSWORD='REPLICA', MASTER_LOG_FILE='MYSQL-BIN.000005',
MASTER_LOG_POS=107;
```

Si slave_IO_Running y SQL_Running están en yes significa que la replicación y configurarición es exitosa

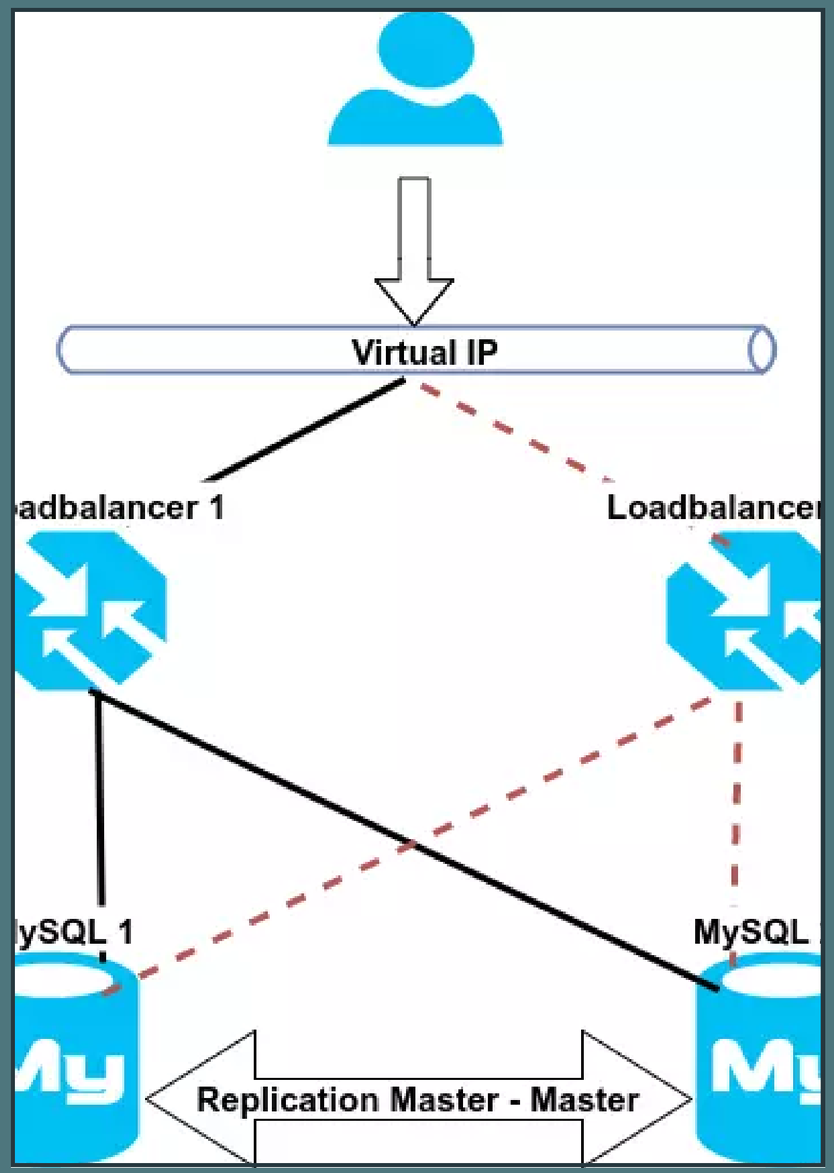
```
vagrant@db2:~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 515
Server version: 5.5.62-0ubuntu0.14.04.1-log (Ubuntu)

Copyright (c) 2000, 2018, 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> show slave status \G
*************************** 1. row ***************************
               Slave_IO_State: Waiting for master to send event
                  Master_Host: 192.168.56.102
                  Master_User: replica
                  Master_Port: 3306
                Connect_Retry: 60
              Master_Log_File: mysql-bin.000005
          Read_Master_Log_Pos: 107
               Relay_Log_File: mysql-relay-bin.000017
                Relay_Log_Pos: 253
        Relay_Master_Log_File: mysql-bin.000005
             Slave_IO_Running: Yes
            Slave_SQL_Running: Yes
              Replicate_Do_DB:
          Replicate_Ignore_DB:
```

DB1(MAESTRO)

```
mysql> select * from test;
+--------------+
| calificacion |
+--------------+
|            4 |
|            2 |
|           85 |
|           10 |
|           60 |
|           23 |
|           55 |
|           10 |
+--------------+
8 rows in set (0.00 sec)
```

DB2(ESCLAVO)

```
vagrant@db3:~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 554
Server version: 5.5.62-0ubuntu0.14.04.1-log (Ubuntu)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All r

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

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

mysql> use testdb
Reading table information for completion of table and column
You can turn off this feature to get a quicker startup with -

Database changed
mysql> select * from test;
+--------------+
| calificacion |
+--------------+
|            4 |
|            2 |
|           85 |
|           40 |
|           10 |
|           60 |
|           23 |
|           55 |
|           10 |
+--------------+
```



CONDIGURACION HAPROXY

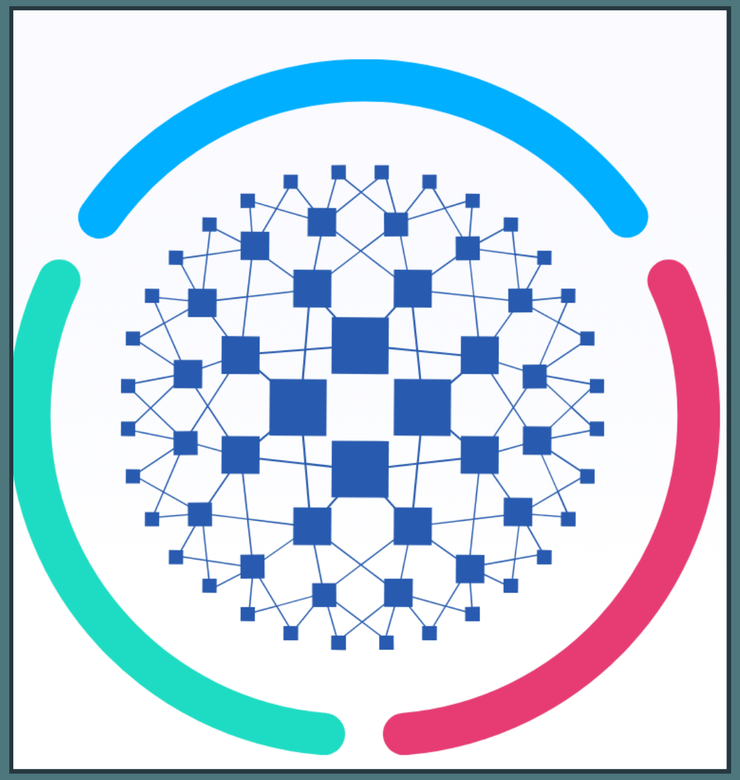
Los siguentes pasos se realizan en maestro y esclavos

```
mysql -uroot -p
USE mysql;
INSERT INTO user (HOST,USER) VALUES ('192.168.56.101',
'haproxy_user');flush privileges;
```

```
GRANT ALL PRIVILEGES ON *.* TO
'HAPROXY_ROOT'@'192.168.56.101' IDENTIFIED BY
'HAPROXY_ROOT_PASS' WITH GRANT
OPTION;FLUSH PRIVILEGES;
```

Por último en el proxy se intala mysql-client

```
SUDO APT-GET INSTALL MYSQL-CLIENT
```



CONFIGURACION HAPROXY

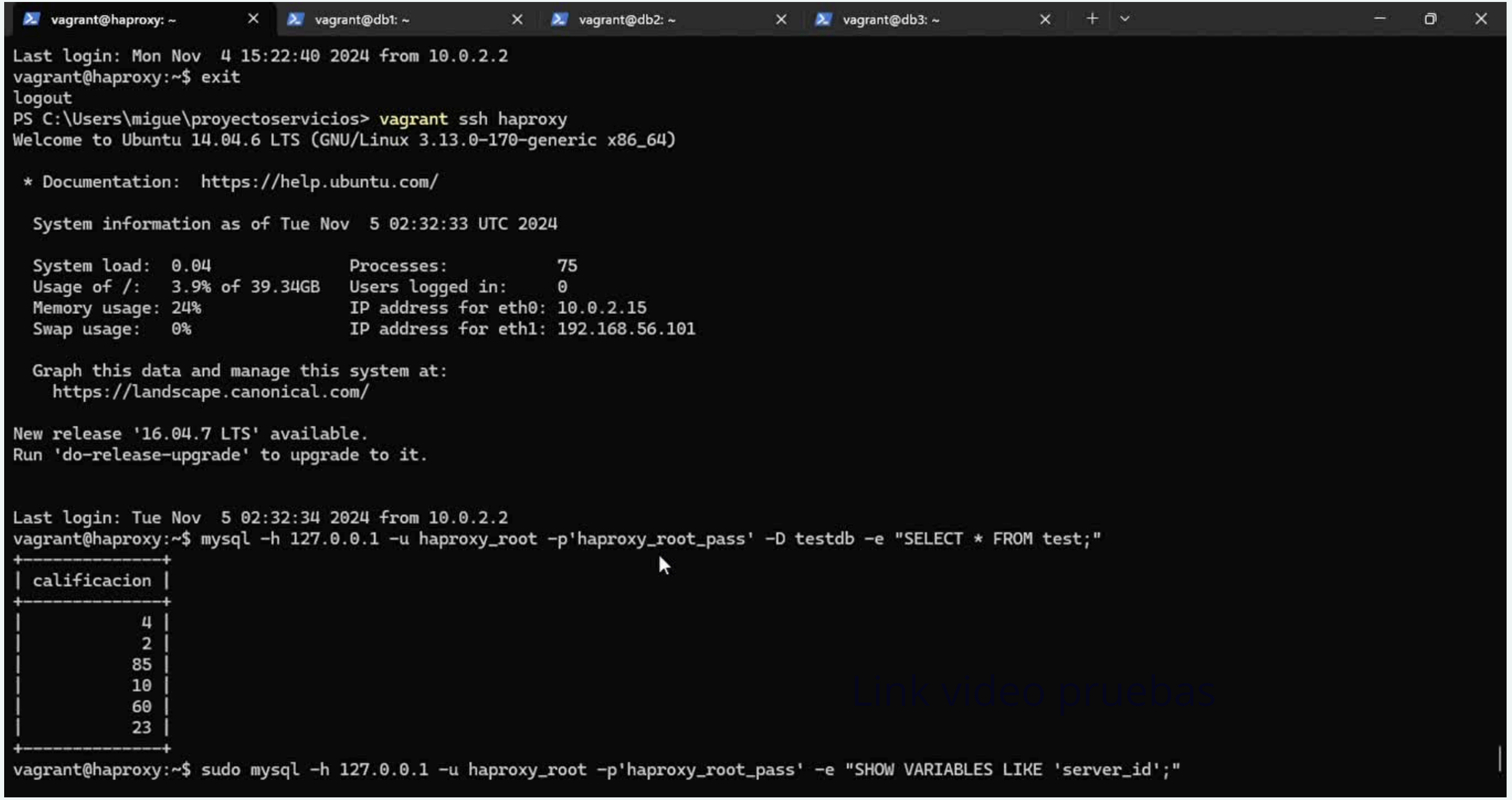
```
 GNU nano 2.2.6                                File: /etc/default/haproxy

# Set ENABLED to 1 if you want the init script to start haproxy.
ENABLED=1
# Add extra flags here.
#EXTRAOPTS="-de -m 16"
```

/ETC/HAPROXY/HAPROXY.CFG

```
global
    log 127.0.0.1 local0
    chroot /var/lib/haproxy
    maxconn 100
    user haproxy
    group haproxy

defaults
    log global
    option tcplog
    retries 2
    timeout client 30m
    timeout connect 4s
    timeout server 30m
    timeout check 5s

listen stats
    mode http
    bind *:9201
    stats enable
    stats uri /stats
    stats realm Strictly\ Private
    stats auth kanyi:test

listen mysql-cluster
    bind :3306
    mode tcp
    option mysql-check user haproxy_user
    balance roundrobin
    server master 192.168.56.102:3306 check
    server slave1 192.168.56.103:3306 check
    server slave2 192.168.56.104:3306 check
```

```
vagrant@haproxy:~$ mysql -h 127.0.0.1 -u haproxy_root -p'haproxy_root_pass' -D testdb -e "SELECT * FROM test;"
+--------------+
| calificacion |
+--------------+
|            4 |
|            2 |
|           85 |
|           10 |
|           60 |
|           23 |
|           55 |
|           10 |
+--------------+
vagrant@haproxy:~$ sudo mysql -h 127.0.0.1 -u haproxy_root -p'haproxy_root_pass' -e "SHOW VARIABLES LIKE 'server_id';"
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| server_id     | 2     |
+---------------+-------+
vagrant@haproxy:~$ sudo mysql -h 127.0.0.1 -u haproxy_root -p'haproxy_root_pass' -e "SHOW VARIABLES LIKE 'server_id';"
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| server_id     | 3     |
+---------------+-------+
vagrant@haproxy:~$ sudo mysql -h 127.0.0.1 -u haproxy_root -p'haproxy_root_pass' -e "SHOW VARIABLES LIKE 'server_id';"
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| server_id     | 1     |
+---------------+-------+
```

PRUEBAS



Link video pruebas

DISCUSION DE PRUEBAS

```
OLTP test statistics:
    queries performed:
        read:                            17850
        write:                           6375
        other:                           2545
        total:                           26770
    transactions:                        1270   (21.12 per sec.)
    deadlocks:                           5      (0.08 per sec.)
    read/write requests:                 24225  (402.88 per sec.)
    other operations:                    2545   (42.33 per sec.)

Test execution summary:
    total time:                          60.1296s
    total number of events:              1270
    total time taken by event execution: 239.8802
    per-request statistics:
         min:                                  125.75ms
         avg:                                  188.88ms
         max:                                  416.44ms
         approx.  95 percentile:               231.36ms

Threads fairness:
    events (avg/stddev):           317.5000/28.02
    execution time (avg/stddev):   59.9700/0.05
```

4 USERS

```
OLTP test statistics:
    queries performed:
        read:                            18116
        write:                           6470
        other:                           2579
        total:                           27165
    transactions:                        1285   (21.34 per sec.)
    deadlocks:                           9      (0.15 per sec.)
    read/write requests:                 24586  (408.30 per sec.)
    other operations:                    2579   (42.83 per sec.)

Test execution summary:
    total time:                          60.2156s
    total number of events:              1285
    total time taken by event execution: 480.3869
    per-request statistics:
         min:                                  266.30ms
         avg:                                  373.84ms
         max:                                 1123.54ms
         approx.  95 percentile:               446.84ms

Threads fairness:
    events (avg/stddev):           160.6250/6.34
    execution time (avg/stddev):   60.0484/0.07
```

8 USERS

```
OLTP test statistics:
    queries performed:
        read:                            14966
        write:                           5345
        other:                           2085
        total:                           22396
    transactions:                        1016   (16.80 per sec.)
    deadlocks:                           53     (0.88 per sec.)
    read/write requests:                 20311  (335.85 per sec.)
    other operations:                    2085   (34.48 per sec.)

Test execution summary:
    total time:                          60.4761s
    total number of events:              1016
    total time taken by event execution: 964.7802
    per-request statistics:
         min:                                  499.82ms
         avg:                                  949.59ms
         max:                                 6013.82ms
         approx.  95 percentile:              1098.53ms

Threads fairness:
    events (avg/stddev):           63.5000/2.45
    execution time (avg/stddev):   60.2988/0.13
```

16 USERS

DISCUSION DE PRUEBAS

HAProxy version 1.4.24, released 2013/06/17

Statistics Report for pid 1239

> General process information

pid = 1239 (process #1, nbproc = 1)
uptime = 0d 0h48m00s
system limits: memmax = unlimited; ulimit-n = 215
maxsock = 215; **maxconn** = 100; **maxpipes** = 0
current conns = 2; current pipes = 0/0
Running tasks: 1/5

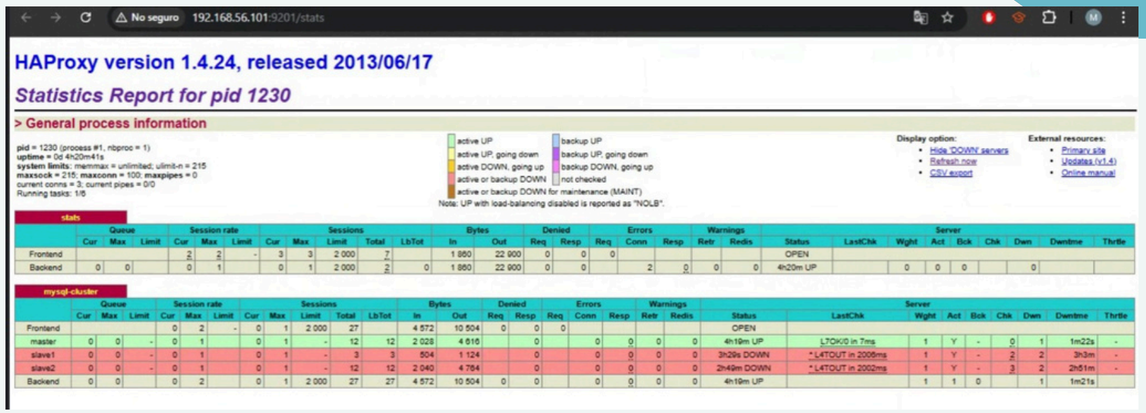
active UP
active UP, going down
active DOWN, going up
active or backup DOWN
active or backup DOWN for mainten
backup
backup
backup
not che
Note: UP with load-balancing disabled is

stats

	Queue			Session rate			Sessions					Bytes		Denied	
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	In	Out	Req	Resp
Frontend				2	2	-	2	2	2 000	7		2 299	23 503	0	
Backend	0	0		0	1		0	1	2 000	2	0	2 299	23 503	0	

mysql-cluster

	Queue			Session rate			Sessions					Bytes		Denied	
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	In	Out	Req	Resp
Frontend				0	12	-	0	16	2 000	78		4 374 111	24 503 154	0	0
master	0	0	-	0	4		0	5	-	26	26	2 439 665	9 444 826		0
slave1	0	0	-	0	4		0	5	-	26	26	547 194	7 354 203		0
slave2	0	0	-	0	4		0	6	-	26	26	1 387 252	7 704 125		0
Backend	0	0		0	12		0	16	2 000	78	78	4 374 111	24 503 154	0	0



No seguro 192.168.56.101:9201/stats

HAProxy version 1.4.24, released 2013/06/17

Statistics Report for pid 1230

> General process information

pid = 1230 (process #1, nbproc = 1)
uptime = 0d 4h20m41s
system limits: memmax = unlimited; ulimit-n = 215
maxsock = 215; maxconn = 100; maxpipes = 0
current conns = 3; current pipes = 0/0
Running tasks: 1/5

active UP
active UP, going down
active DOWN, going up
active or backup DOWN
active or backup DOWN for maintenance (MAINT)
backup UP
backup UP, going down
backup DOWN, going up
not checked
Note: UP with load-balancing disabled is reported as "NOLB".

Display option:
- Hide 'DOWN' servers
- Refresh now
- CSV export

External resources:
- Primary site
- Updates (v1.4)
- Online manual

stats

	Queue			Session rate			Sessions					Bytes		Denied		Errors			Warnings		Server								
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
Frontend				2	2	-	3	3	2 000	7		1 860	22 900	0	0	0					OPEN								
Backend	0	0		0	1		0	1	2 000	2	0	1 860	22 900	0	0	0		2	0	0	0	4h20m UP		0	0	0		0		

mysql-cluster

	Queue			Session rate			Sessions					Bytes		Denied		Errors			Warnings		Server								
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
Frontend				0	2	-	0	1	2 000	27		4 572	10 504	0	0	0					OPEN								
master	0	0	-	0	1		0	1	-	12	12	2 028	4 616		0		0	0	0	0	4h19m UP	L7OK/0 in 7ms	1	Y	-	0	1	1m22s	-
slave1	0	0	-	0	1		0	1	-	3	3	504	1 124		0		0	0	0	0	3h29s DOWN	* L4TOUT in 2006ms	1	Y	-	2	2	3h3m	-
slave2	0	0	-	0	1		0	1	-	12	12	2 040	4 764		0		0	0	0	0	2h49m DOWN	* L4TOUT in 2002ms	1	Y	-	3	2	2h51m	-
Backend	0	0		0	2		0	1	2 000	27	27	4 572	10 504	0	0		0	0	0	0	4h19m UP		1	1	0		1	1m21s	

Note: UP with load-balancing disabled is reported as "NOLB".

stats

	Queue			Session rate			Sessions					Bytes		Denied		Errors			Warnings		Server								
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
Frontend				1	2	-	2	2	2 000	3		439	260	0	0	0					OPEN								
Backend	0	0		0	0		0	0	2 000	0	0	439	260	0	0		0	0	0	0	36m37s UP		0	0	0		0		

mysql-cluster

	Queue			Session rate			Sessions					Bytes		Denied		Errors			Warnings		Server								
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
Frontend				0	2	-	0	1	2 000	4		674	1 521	0	0	0					OPEN								
master	0	0	-	0	1		0	1	-	2	2	334	727		0		0	0	0	0	35m14s UP	L7OK/0 in 7ms	1	Y	-	0	1	1m21s	-
slave1	0	0	-	0	1		0	1	-	1	1	170	397		0		0	0	0	0	34m5s UP	L7OK/0 in 4ms	1	Y	-	0	1	2m30s	-
slave2	0	0	-	0	1		0	1	-	1	1	170	397		0		0	0	0	0	32m47s UP	L7OK/0 in 7ms	1	Y	-	2	2	3m34s	-
Backend	0	0		0	2		0	1	2 000	4	4	674	1 521	0	0		0	0	0	0	35m14s UP		3	3	0		1	1m20s	

MUCHAS GRACIAS