

REDIS - ADMINISTRACIÓN

¿QUÉ VAMOS APRENDER?

1. Redis Aspectos Generales.
2. Instalación
3. Redis-Simple Instancia.
4. Server Configuración
5. Redis Múltiple instancia-Master/Esclavo-Replicación.
6. Redis Múltiple instancia-Cluster.

ASPECTOS GENERALES

¿QUÉ ES REDIS ?

Es un almacén de estructura de datos en memoria, usado como base de datos y cache. Soporta estructura de datos tales como: String, hashed, list, sets, sorted sets. Soporta operaciones atómicas sobre estas estructuras de datos tales como agregar data a String, incrementar un valor en un hash, agregar un elemento a una lista, y fijar operaciones como inserción, unión y diferencia.

Redis tiene un proceso de replicación incorporada, LUA Scripting y transacciones y niveles de persistencia en disco, y proporciona niveles de alta disponibilidad.

- Nos da acceso rápido a los datos NoSQL.
- Mejora los tiempos de respuesta de su aplicación con una infraestructura mínima. (interés de twitter)
- Proporciona Alta Disponibilidad, clustering , persistencia y opciones de seguridad.

Usado por compañías como: Twitter, Pinterest, StackOverflow , etc

BONDADES DE REDIS.

- Redis almacena un par de datos en memoria Clave – Valor.
- Clave – Valor, son almacenados en la memoria primaria (RAM).
- Las bases de datos tradicionales almacenan la data en memoria secundaria(disk).
- La memoria primara es mucho más rápido, pero más costosa.
- El inconveniente es que Redis no puede ser usado para almacenar archivos grandes. Pero en vez de ello se basa en el acceso rápido a los conjuntos de datos más pequeños.
- Altamente escalable.

REDIS - INSTALACIÓN

```
#Revisando las configuraciones por defecto
$ cd /opt/
$ tar vxf redislabs-5.0.2-15-rhel7-x86_64.tar
$ ./install.sh -s /var/run/redislabs
```

```
2018-04-16 09:01:48 [.] Checking root access
2018-04-16 09:01:48 [!] Running as user root, sudo is not required.
2018-04-16 09:01:49 [.] Appending to paths.sh
2018-04-16 09:01:49 [.] Creating socket directory /var/run/redislabs
2018-04-16 09:01:49 [.] Writing socket directory to /etc/opt/redislabs/config.js
on
2018-04-16 09:01:50 [?] Swap is enabled. Do you want to proceed? [Y/N]? y
```

Transaction Summary

=====

Install 2 Packages

Total size: 164 M

Installed size: 164 M

Downloading packages:

Running transaction check

Running transaction test

Transaction test succeeded

Running transaction

Installing : redislabs-5.0.2-15.rhel7.x86_64 1/2

Firewalld is not running

Installing : redislabs-utils-5.0.2-15.rhel7.x86_64 2/2

Verifying : redislabs-5.0.2-15.rhel7.x86_64 1/2

Verifying : redislabs-utils-5.0.2-15.rhel7.x86_64 2/2

Installed:

redislabs.x86_64 0:5.0.2-15.rhel7 redislabs-utils.x86_64 0:5.0.2-15.rhel7

Complete!

2018-04-16 09:03:31 [?] Do you want to automatically tune the system for best performance [Y/N]? █


```
2018-04-16 09:06:45 [.] Removing systune from rc.local if necessary.
2018-04-16 09:06:45 [?] Cluster nodes must have their system time synchronized.
Do you want to set up NTP time synchronization now [Y/N]? y
2018-04-16 09:08:37 [.] Making sure NTP is installed and time is set.
2018-04-16 09:08:38 [!] Chrony service is already installed, skipping NTP installation.
2018-04-16 09:08:38 [$] executing: 'systemctl enable chronyd.service'
Created symlink from /etc/systemd/system/multi-user.target.wants/chronyd.service to /usr/lib/systemd/system/chronyd.service.
2018-04-16 09:08:38 [$] executing: 'systemctl start chronyd.service'
2018-04-16 09:08:38 [$] executing: 'chronyc -a makestep'
200 OK
2018-04-16 09:08:38 [?] This machine seems to have a firewall installed.
Would you like to open the ports utilized by Redis Enterprise on this machine's default firewall zone [Y/N]? y
2018-04-16 09:08:42 [$] executing: '/bin/firewall-cmd --add-service=redislabs'
Firewalld is not running
2018-04-16 09:08:42 [$] executing: '/bin/firewall-cmd --add-service=redislabs --permanent'
Firewalld is not running
2018-04-16 09:08:43 [!] Note: Log files will be stored on the root file system, in path /var/opt/redislabs/log

RedisLabs rest-api documentation has been deployed in /usr/share/doc/redislabs .

2018-04-16 09:08:44 [!] Installation is complete!
2018-04-16 09:08:44 [?] Would you like to run rlcheck to verify proper configuration? [Y/N]? y
2018-04-16 09:08:45 [$] executing: '/opt/redislabs/bin/rlcheck --suppress-tests=verify_bootstrap_status,verify_processes,verify_pidfiles'
Saving to file: /var/opt/redislabs/log/rlcheck.log
(Remark : Will stop on first failure)
##### Welcome to RedisLabs Enterprise Cluster settings verification utility #####
Skipping test: verify_bootstrap_status
Skipping test: verify_processes
Running test: verify_dmcproxy
PASS
Running test: verify_port_range
PASS
Skipping test: verify_pidfiles
Running test: verify_capabilities
PASS
Running test: verify_existing_sockets
PASS

Summary:
-----
ALL TESTS PASSED.

2018-04-16 09:08:46 [$] executing: 'chown redislabs:redislabs /var/opt/redislabs/log/rlcheck.log'
2018-04-16 09:08:46 [!] Please logout and login again to make sure all environment changes are applied
```

Summary:

ALL TESTS PASSED.

2018-04-16 09:08:46 [\$] executing: 'chown redislabs:redislabs /var/opt/redislabs/log/rlcheck.log'

2018-04-16 09:08:46 [!] Please logout and login again to make sure all environment changes are applied.

2018-04-16 09:08:46 [!] Point your browser at the following URL to continue:

2018-04-16 09:08:46 [!] <https://192.168.1.109:8443>

2018-04-16 09:08:46 [\$] executing: 'chmod 644 /tmp/install.log'

2018-04-16 09:08:46 [\$] executing: 'chown redislabs:redislabs /tmp/install.log'

2018-04-16 09:08:46 [!] Calling leash.py off

Cluster is not ready

[root@stationINS opt]#

redis enterprise

by redis labs

Version 5.0.2-15

Setup

© 2018 Redis Labs, Inc. All rights reserved.

node configuration

Persistent storage path	<input type="text" value="/var/opt/redislabs/persist"/>	Free = 23.32 GB																		
Ephemeral storage path ⓘ	<input type="text" value="/var/opt/redislabs/tmp"/>	Free = 23.32 GB																		
<input type="checkbox"/> Enable flash storage support ⓘ Read more																				
IP Addresses Usage ⓘ Read more	<table><thead><tr><th>IP Address</th><th>Interface</th><th>Internal traffic ⓘ</th><th>External traffic ⓘ</th></tr></thead><tbody><tr><td colspan="4">IPv4</td></tr><tr><td>192.168.1.109</td><td>ens33</td><td><input checked="" type="radio"/></td><td><input checked="" type="checkbox"/></td></tr><tr><td colspan="4"><div>+</div></td></tr></tbody></table>				IP Address	Interface	Internal traffic ⓘ	External traffic ⓘ	IPv4				192.168.1.109	ens33	<input checked="" type="radio"/>	<input checked="" type="checkbox"/>	<div>+</div>			
IP Address	Interface	Internal traffic ⓘ	External traffic ⓘ																	
IPv4																				
192.168.1.109	ens33	<input checked="" type="radio"/>	<input checked="" type="checkbox"/>																	
<div>+</div>																				
Cluster configuration	<div><input checked="" type="radio"/> Create new cluster <input type="radio"/> Join cluster</div> <div>Cluster name (FQDN) Read more <input type="text" value="mycluster"/></div> <div><input type="checkbox"/> Enable private & public endpoints support Read more <input type="checkbox"/> Enable rack-zone awareness Read more</div>																			

Next

cluster authentication

If you purchased a cluster key please insert it in the field below and click Next.
If you would like to use the free version just click Next.

Cluster key

By clicking "Next" you agree to the [terms and conditions](#) associated with this product.

Back Next

redis enterprise | by redis labs

set admin credentials

Email

Password

Verify password

Back

Next

○ please wait...

Creating cluster...

redis enterp

cluster nodes da

gonzales@edu.com.pe



documentation Support Sign Out

create new database



redis database



memcached database

Runs on

RAM ▼

Deployment

Single Region ▼

Next

create database

Name	<input type="text" value="redis1db"/>
Protocol	Redis
Runs on	RAM
Memory limit (GB) Read more	<input type="text" value="0.1"/> GB 1.37 GB RAM unallocated
<input type="checkbox"/> Replication i	
<input type="checkbox"/> Redis Modules	None
Data persistence	<input type="text" value="None"/> ▼
Redis password	<input type="password"/> 🔑
Endpoint port number	<input type="text" value="14000"/>
<input type="checkbox"/> Database clustering Read more	
Data eviction policy i	<input type="text" value="volatile-lru"/> ▼
<input type="checkbox"/> Replica of i Read more	
<input type="checkbox"/> SSL Authentication	
<input type="checkbox"/> Periodic backup	

☐ Replica of ⓘ
Read more

☐ SSL Authentication

☐ Periodic backup

Alerts ⓘ

<input type="checkbox"/> Dataset size has reached	<input type="text"/>	% of the memory limit
<input type="checkbox"/> Throughput is higher than	<input type="text"/>	RPS (requests per second)
<input type="checkbox"/> Throughput is lower than	<input type="text"/>	RPS (requests per second)
<input type="checkbox"/> Latency is higher than	<input type="text"/>	msec
<input type="checkbox"/> Periodic backup has been delayed for longer than	<input type="text"/>	minutes

☐ Receive email alerts ⓘ

Cancel

Activate

[Hide advanced options](#)

database

redis1db ▼



metrics slowlog configuration

Activated on	4/15/2018 11:45:44 PM
Last changed	4/15/2018 11:45:44 PM
Protocol	Redis
Runs on	RAM
Endpoint ⓘ Get Replica of source URL	redis-14000.mycluster:14000 / 192.168.1.109:14000
Version	Redis version compliance 4.0.2
Memory	Memory limit 0.10 GB Used memory
Replication	Disabled
Redis Modules	None
Persistence	None
Redis password	
Database clustering	None

Google Traductor

Installing the Package | R

Enterprise-Class Redis for

← → ↺

No es seguro | https://192.168.1.109:8443/#/bdb/tabs/conf/1

☆ ⋮

Memory	Memory limit 0.10 GB Used memory 1.8 MB
Replication	Disabled
Redis Modules	None
Persistence	None
Redis password	
Database clustering	None
Eviction policy	volatile-lru
Replica of Read more	None
SSL Authentication	None
Periodic backup	None
Alerts	None

←

Edit

Delete

Import

Export

Windows Taskbar

System Tray

18:46
15/04/2018

REDIS - ARQUITECTURA SIMPLE INSTANCIA

SIMPLE INSTANCIA

En la simple instancia existe un redis cliente que se comunica a in redis server que contiene almacén de data.

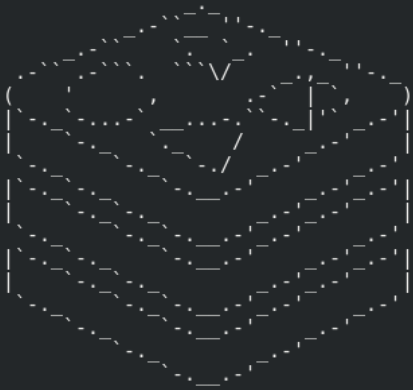


INSTALANDO y VALIDANDO

```
$wget http://download.redis.io/redis-stable.tar.gz
$tar xvzf redis-stable.tar.gz
$cd redis-stable
$make
```

.....desempaquetando elementos

```
$make install
$redis-server &
```



Redis 4.0.8 (00000000/0) 64 bit

Running in standalone mode
Port: 6379
PID: 2231

<http://redis.io>

```
$redis-cli
172.0.0.1:6379> ping
PONG
172.0.0.1:6379>
```

#Vamos a ver las carateristicas de nuestro server

```
$redis-cli
```

```
>INFO
```

```
#server .....
```

```
#Clients .....
```

```
#Memory .....
```

```
#Persistence .....
```

```
#Stats .....
```

#Para salir del server

```
172.0.0.1:6379>quit
```

#Para matar el proceso de server desde el cliente

```
$redis-cli shutdown
```

#COMANDOS DE USO GENERAL

#para conectarse y ejecutar comandos use:

```
$redis-cli
```

#eliminando todo el contenido de la base de datos

```
>flushall
```

#Trabajando en un simple instancia

```
>set taco:1000 "Hola como estas"
```

```
>get taco:1000
```

```
>flushall
```

#para salir de redis use quit

```
>quit
```

#para limpiar console

```
>clear
```

#Para matar el proceso de server

```
$redis-cli shutdown
```

REDIS - ADMINISTRACIÓN Y SEGURIDAD

REDIS - ADMINISTRACIÓN Y SEGURIDAD

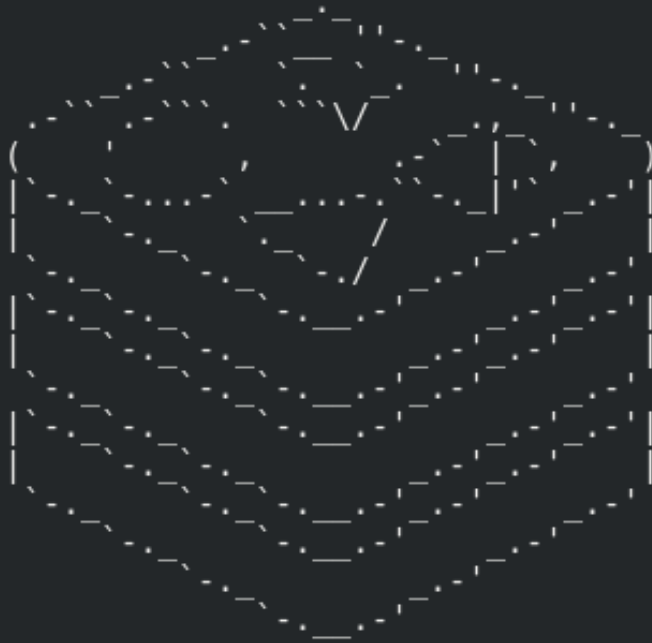
CONFIGURACIONES

#Revisando las configuraciones por defecto

\$ redis-server

```
2231:U 12 Apr 22:35:11.125 # warning: no config file specified, using the default config. In order to specify  
a config file use redis-server /path/to/redis.conf
```

```
2231:M 12 Apr 22:35:11.127 * Increased maximum number of open files to 10032 (it was originally set to 1024).
```



```
Redis 4.0.8 (00000000/0) 64 bit
```

```
Running in standalone mode
```

```
Port: 6379
```

```
PID: 2231
```

```
http://redis.io
```

```
2231:M 12 Apr 22:35:11.130 # WARNING: The TCP backlog setting of 511 cannot be enforced because /proc/sys/net  
/core/somaxconn is set to the lower value of 128.
```

```
2231:M 12 Apr 22:35:11.130 # Server initialized
```

```
2231:M 12 Apr 22:35:11.130 # WARNING: The TCP backlog setting of 511 cannot be enforced because /proc/sys/net
```

#levantando el server especificando otro puerto en el archivo de configuración por ejemplo port 6378

\$ vim redis.conf

#Laod el server especificando un archivo de entrada para las configuraciones.

```
$redis-server redis.conf
```

#Laod una instancia redis pasándole parámetro para definir un nuevo puerto.

```
$redis-server --port 6380
```

cargando el cliente redis pero pasándole por parámetro el puerto del server a donde tengo que conectarme

```
$redis-cli -p 6380
```

```
>quit
```

#Ud puede hacer cambios en caliente sobre la marcha sobre algunos parámetros..por ejemplo cambiemos el slowlog-max-len

#revise las configuraciones para ese parámetro en el archivo redis.conf

```
$vim redis.conf
```

```
$redis-cli
```

```
#una ves dentro cambie en caliente el valor del parametro  
>CONFIG SET slowlog-max-len 129  
>quit
```

REDIS - ADMINISTRACIÓN Y SEGURIDAD

SEGURIDAD

#SEGURIDAD - Configurando para que nuestro server solicite password de los clientes que quieran conectarse.

\$redis-cli

#ejecute un set de registro

>set x 1

#luego lance el comando

>CONFIG SET requirepass mysuperpass

Ok

#vuelvo a usar el set

>set x 2

(error)NOAUTH Authentication required.

#autenticase con el siguiente comando

>AUTH mysuperpass

OK

#Vuelva a lanzar el comando set

>set x 1

Ok

>quit

\$

REFERENCIAS LINK

- <https://redis.io/topics/security>
- <https://redis.io/>
 - documentation
 - Administration
 - Security

REDIS - ADMINISTRACIÓN Y SEGURIDAD

LOGGING Y MONITOREO

#Revisando las características de nuestro server instanciado

\$redis-cli

>INFO

```
# Server
redis_version:4.0.8
redis_git_sha1:00000000
redis_git_dirty:0
redis_build_id:fc91a11a5450cb30
redis_mode:standalone
os:Linux 4.10.14-200.fc25.x86_64 x86_64
arch_bits:64
multiplexing_api:epoll
atomicvar_api:atomic-builtin
gcc_version:6.3.1
process_id:3323
run_id:6bca6dc7bcbd6d1fe56fc129997b0939dcb87678
tcp_port:6379
uptime_in_seconds:2232
uptime_in_days:0
hz:10
lru_clock:13851248
executable:/walter/redis-4.0.8/redis-server
config_file:/walter/redis-4.0.8/redis.conf
```

```
# Clients
connected_clients:1
client_longest_output_list:0
client_biggest_input_buf:0
blocked_clients:0
```

#Puede fijar una búsqueda para una sección específica

\$redis-cli -p 6380

>INFO server

#Vamo a monitorear la peticiones que se generan sobre el server con el cliente

```
$redis-cli
```

```
>MONITOR
```

Ok

#Luego abrir otro terminal y conectarse al server

```
$redis-cli -p 6380
```

```
>set x 1
```

```
>info server
```

#vaya a la ventana donde se lanzo el monitor y valida el login de los eventos.

```
231231233333.2323 [127.0.0.1:38050] "set" "x" "1"
```

```
231231233333.2323 [127.0.0.1:38050] command
```

#Para conseguir la información del log ejecutemos el siguiente comando.

```
$redis-cli -p 6380
```

```
>SLOWLOG GET
```

#Para ver el tamaño dl log

```
>SLOWLOG LEN
```

#Para el reset de la información gestionada por el log

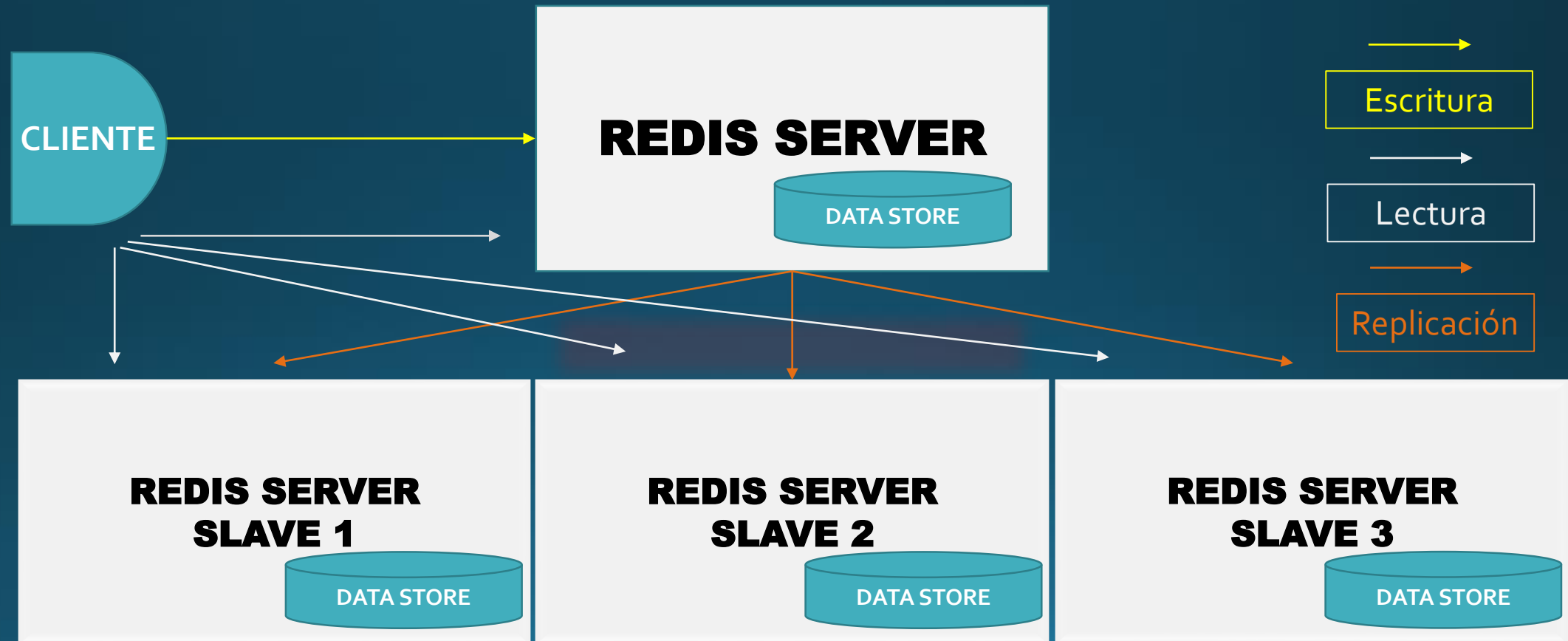
```
>SLOWLOG reset
```

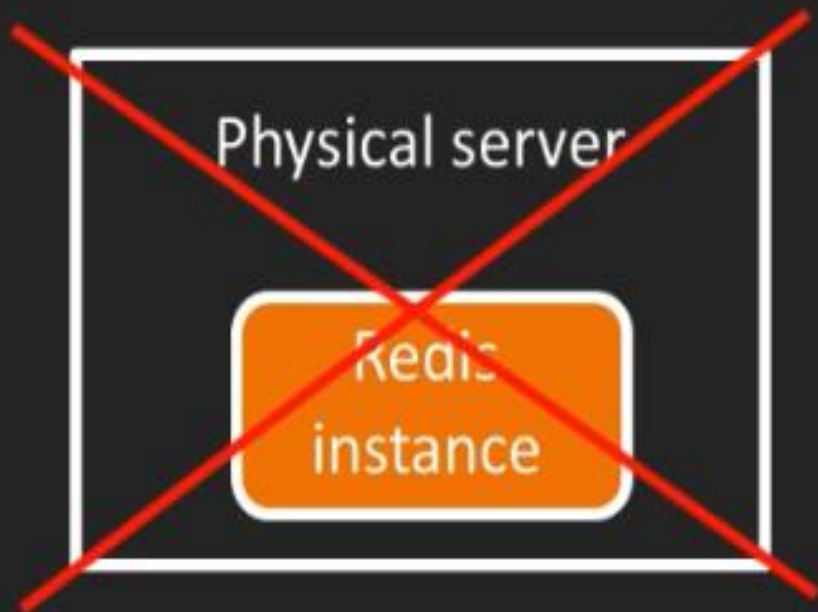
REDIS - Redis Múltiple instancia Master/Esclavo-Replicación

SCALING AND HIGH AVAILABILITY

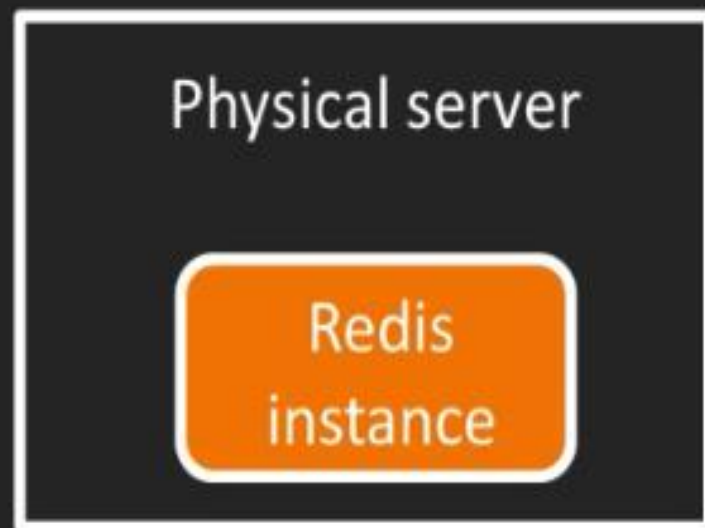
MULTIPLE INSTANCIA MASTER/SLAVE

1. Cliente escribe sobre el master
2. Redis master informa a los esclavos a través de la replicación(1:muchos) copiando así la misma y toda la data en todas las instancias server (db)
3. Si el cliente quiere leer lo puede hacer a cualquier nodo master o esclavos





...

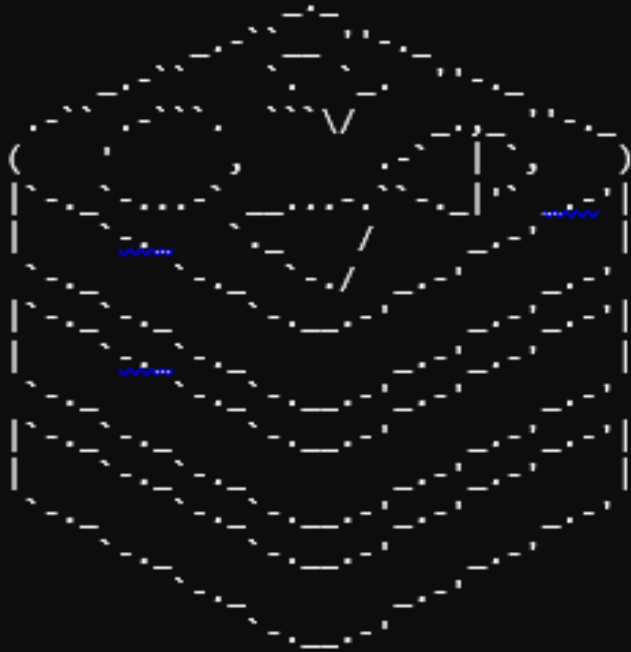


#Vamos a configurar en una sola instalación varias instancias para simular nuestra infraestructura master esclavo.

#instancia 1

\$redis-server --port 6380 --dbfilename db1.db

```
2102:C 15 Apr 18:04:07.485 # o000o000o000o Redis is starting o000o000o000o
2102:C 15 Apr 18:04:07.485 # Redis version=4.0.8, bits=64, commit=00000000,
modified=0, pid=2102, just started
2102:C 15 Apr 18:04:07.485 # Configuration loaded
2102:M 15 Apr 18:04:07.487 * Increased maximum number of open files to 10032 (it
was originally set to 1024).
```



Redis 4.0.8 (00000000/0) 64 bit

Running in standalone mode

Port: 6380

PID: 2102

<http://redis.io>

#Vamos a configurar en una sola instalación varias instancias para simular nuestra infraestructura master esclavo.

#instancia 1

\$redis-server --port 6380 --dbfilename db1.db

```
2102:M 15 Apr 18:04:07.489 # WARNING: The TCP backlog setting of 511 cannot be enforced because
/proc/sys/net/core/somaxconn is set to the lower value of 128.2102:M 15 Apr 18:04:07.490 # Server initialized2102:M 15 Apr
18:04:07.490 # WARNING overcommit_memory is set to 0! Background save may fail under low memory condition. To fix this
issue add 'vm.overcommit_memory = 1' to /etc/sysctl.conf and then reboot or run the command 'sysctl vm.overcommit_memory=1'
for this to take effect.2102:M 15 Apr 18:04:07.490 # WARNING you have Transparent Huge Pages (THP) support enabled in your
kernel. This will create latency and memory usage issues with Redis. To fix this issue run the command 'echo never >
/sys/kernel/mm/transparent_hugepage/enabled' as root, and add it to your /etc/rc.local in order to retain the setting after
a reboot. Redis must be restarted after THP is disabled.2102:M 15 Apr 18:04:07.490 * Ready to accept connections
```


#Vamos a configurar en una sola instalación varias instancias para simular nuestro infraestructura master esclavo.

```
$redis-server --port 6381 --dbfilename db2.db
```

```
2184:C 15 Apr 18:08:04.885 # o000o000o000o Redis is starting o000o000o000o
2184:C 15 Apr 18:08:04.885 # Redis version=4.0.8, bits=64, commit=00000000, modified=0, pid=2184,
just started2184:C 15 Apr 18:08:04.885 # Configuration loaded
2184:M 15 Apr 18:08:04.887 * Increased maximum number of open files to 10032 (it was originally set
to 1024).
```



#Vamos a configurar en una sola instalación varias instancias para simular nuestra infraestructura master esclavo.

#instancia 2

```
2184:M 15 Apr 18:08:04.888 # WARNING: The TCP backlog setting of 511 cannot be enforced because
/proc/sys/net/core/somaxconn is set to the lower value of 128.2184:M 15 Apr 18:08:04.888 # Server
initialized2184:M 15 Apr 18:08:04.888 # WARNING overcommit_memory is set to 0! Background save may
fail under low memory condition. To fix this issue add 'vm.overcommit_memory = 1' to
/etc/sysctl.conf and then reboot or run the command 'sysctl vm.overcommit_memory=1' for this to
take effect.2184:M 15 Apr 18:08:04.888 # WARNING you have Transparent Huge Pages (THP) support
enabled in your kernel. This will create latency and memory usage issues with Redis. To fix this
issue run the command 'echo never > /sys/kernel/mm/transparent_hugepage/enabled' as root, and add
it to your /etc/rc.local in order to retain the setting after a reboot. Redis must be restarted
after THP is disabled.2184:M 15 Apr 18:08:04.888 * Ready to accept connections
```

#vamos a definir al localhost:6381 como esclavo de localhost:6380

\$redis-cli -p 6381

\$127.0.0.1:6381> SLAVEOF localhost 6380

OK

Salida en el 6381

```
2184:S 15 Apr 18:21:09.641 * Before turning into a slave, using my master parameters to synthesize a
cached master: I may be able to synchronize with the new master with just a partial transfer.2184:S
15 Apr 18:21:09.641 * SLAVE OF localhost:6380 enabled (user request from 'id=2 addr=127.0.0.1:55284
fd=8 name= age=467 idle=0 flags=N db=0 sub=0 psub=0 multi=-1 qbuf=0 qbuf-free=32768 obl=0 oll=0
omem=0 events=r cmd=slaveof')2184:S 15 Apr 18:21:10.288 * Connecting to MASTER localhost:63802184:S
15 Apr 18:21:10.291 * MASTER <-> SLAVE sync started2184:S 15 Apr 18:21:10.291 * Non blocking connect
for SYNC fired the event.2184:S 15 Apr 18:21:10.292 * Master replied to PING, replication can
continue...2184:S 15 Apr 18:21:10.292 * Trying a partial resynchronization (request
ad00e69d8944af0f9d7322212e331198863a0809:1).2184:S 15 Apr 18:21:10.294 * Full resync from master:
e4b6bd0d434ce8442c39a06f318803a350d2ff73:02184:S 15 Apr 18:21:10.294 * Discarding previously cached
master state.2184:S 15 Apr 18:21:10.391 *
MASTER <-> SLAVE sync: receiving 175 bytes from master2184:S 15 Apr 18:21:10.391 *
MASTER <-> SLAVE sync: Flushing old data2184:S 15 Apr 18:21:10.391 *
MASTER <-> SLAVE sync: Loading DB in memory2184:S 15 Apr 18:21:10.391 *
MASTER <-> SLAVE sync: Finished with success
```

```
#vamos a definir al localhos:6381 como esclavo de localhos:6380
```

```
$redis-cli -p 6381
```

```
$127.0.0.1:6381> SLAVEOF localhost 6380
```

OK

Salida en el 6380

2102:M 15 Apr 18:21:10.292 *

```
Slave [::1]:6381 asks for synchronization2102:M 15 Apr 18:21:10.292 * Partial resynchronization not
accepted: Replication ID mismatch (Slave asked for 'ad00e69d8944af0f9d7322212e331198863a0809', my
replication IDs are '2518231afcb0458dab7afcb00daeb06576386f476' and
```

```
'000000000000000000000000000000000000')2102:M 15 Apr 18:21:10.292 * Starting BGSAVE for SYNC with
target: disk2102:M 15 Apr 18:21:10.294 * Background saving started by pid 22952295:C 15 Apr
```

```
18:21:10.343 * DB saved on disk2295:C 15 Apr 18:21:10.344 * RDB: 0 MB of memory used by copy-on-
```

```
write2102:M 15 Apr 18:21:10.390 * Background saving terminated with success2102:M 15 Apr
```

```
18:21:10.391 * Synchronization with slave [::1]:6381 succeeded
```

```
#vamos a definir al localhos:6381 como esclavo de localhos:6380
```

```
$redis-cli -p 6381
```

```
$127.0.0.1:6381> SLAVEOF localhost 6380
```

OK

Salida en el 6380

2102:M 15 Apr 18:21:10.292 *

```
Slave [::1]:6381 asks for synchronization2102:M 15 Apr 18:21:10.292 * Partial resynchronization not
accepted: Replication ID mismatch (Slave asked for 'ad00e69d8944af0f9d7322212e331198863a0809', my
replication IDs are '2518231afcb0458dab7afcb00daeb06576386f476' and
```

```
'000000000000000000000000000000000000')2102:M 15 Apr 18:21:10.292 * Starting BGSAVE for SYNC with
target: disk2102:M 15 Apr 18:21:10.294 * Background saving started by pid 22952295:C 15 Apr
```

```
18:21:10.343 * DB saved on disk2295:C 15 Apr 18:21:10.344 * RDB: 0 MB of memory used by copy-on-
```

```
write2102:M 15 Apr 18:21:10.390 * Background saving terminated with success2102:M 15 Apr
```

```
18:21:10.391 * Synchronization with slave [::1]:6381 succeeded
```

#Trabajando con nuestra arquitectura Master - Slave

```
127.0.0.1:6381> quit
```

```
$redis-cli -p 6380
```

```
127.0.0.1:6380> set master valor
```

```
OK
```

```
127.0.0.1:6380> get master"valor"
```

```
127.0.0.1:6380> quit
```

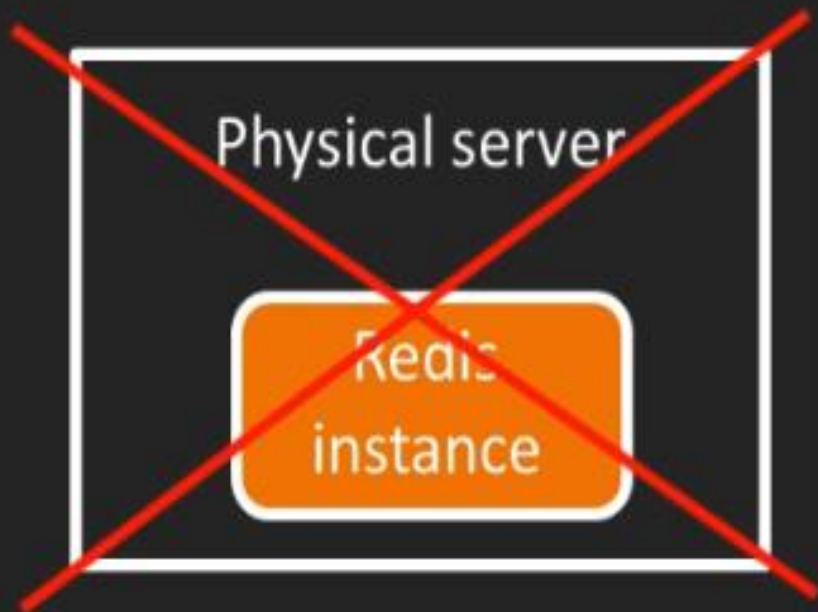
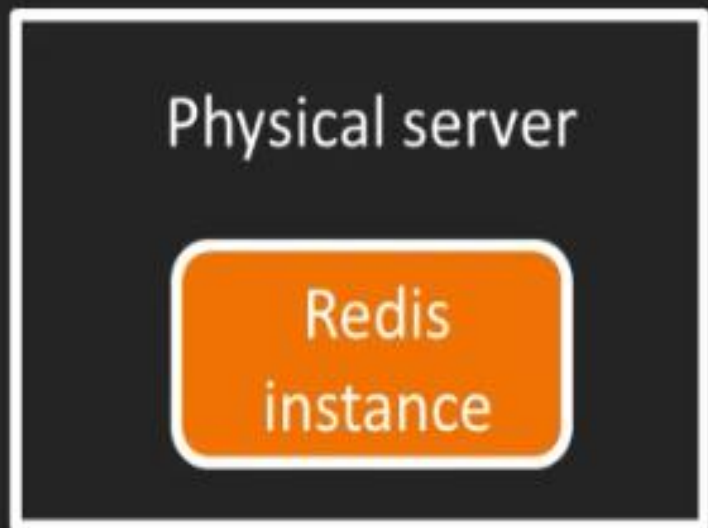
```
$redis-cli -p 6381
```

```
127.0.0.1:6381> get master"valor"
```

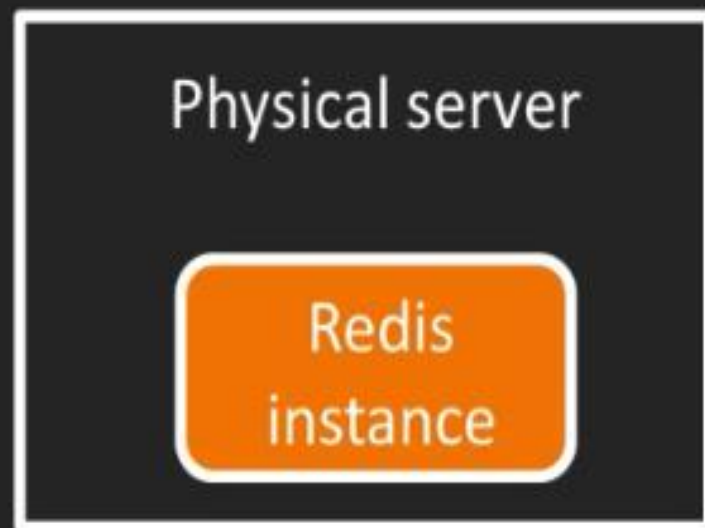
```
127.0.0.1:6381> set slave valor
```

```
(error) READONLY You can't write against a read only slave
```

```
127.0.0.1:6381> quit
```



...



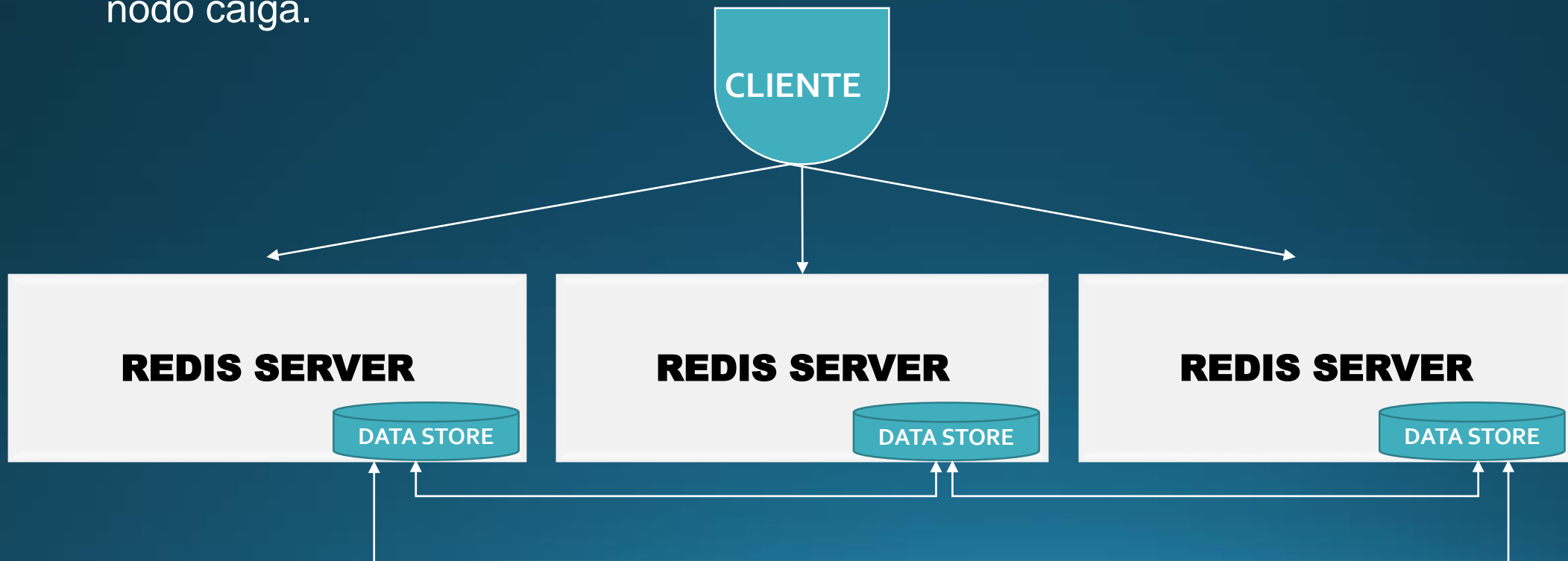
```
#sacando de la regla de juego a nuestro esclavo.. desde el mismo  
#esclavo podemos romper la conexion  
:6381>SLAVEOF NO ONE
```

Como se muestra en el master 6380

```
2102:M 15 Apr 19:05:43.892 # Connection with slave [::1]:6381 lost.
```


MULTIPLE INSTANCIA CLUSTER

1. Cliente se comunica con 1:muchos - redis server
2. Los Redis servers se comunican entre sí compartiendo la data compartida a través de los servers
3. Cada nodo es master y slave a la vez de modo que permita la replicación de la data, garantizando así la disponibilidad de la data aun cuando algún nodo caiga.



BENEFICIOS - MÚLTIPLE INSTANCIA CLUSTER

1. La capacidad de dividir automáticamente el conjunto de datos entre varios nodos.
2. La capacidad de continuar las operaciones cuando un subconjunto de los nodos está teniendo errores o no se puede comunicar con el resto del clúster.
3. Mayor rendimiento: el rendimiento aumenta de manera lineal a medida que aumenta el número de particiones.
4. Mayor tamaño de memoria: aumenta de manera lineal a medida que aumenta el número de particiones.

GRACIAS