Setup of redis

Clément Levallois

2017-04-26

Table of Contents

last modified: 2018-02-04

System

- I use Debian, version 8.7 (why?)
- Vi is used as a text editor in the following

Why Redis?

- I need a data store of the kind: key → multiple values. Millions of keys, with hundreds or thousands of values for each key.
- If in RAM, these queries on key-values than scanning tables in a db - I think.
- Redis offers join / intersection / … operations on sets of keys, which are useful in my use case.

Installing Redis

```
sudo apt-get update && sudo apt-get upgrade
sudo apt-get install software-properties-common
```

DO (source: https://www.linode.com/docs/databases/redis/deploy-redis-on-ubuntu-or-debian)

```
sudo vi /etc/apt/sources.list.d/dotdeb.list
```

```
in this file, write:
deb http://packages.dotdeb.org stable all
deb-src http://packages.dotdeb.org stable all
```

Then: wget https://www.dotdeb.org/dotdeb.gpg sudo apt-key add dotdeb.gpg sudo apt-get update sudo apt-get install redis-server

Stop / start / restart redis:

sudo service redis-server restart

OR sudo wget http://download.redis.io/redis-stable.tar.gz sudo tar xvzf redis-stable.tar.gz cd redis-stable make sudo cp src/redis-server /usr/local/bin/ sudo cp src/redis-cli /usr/local/bin/

```
sudo mkdir /etc/redis
sudo mkdir /var/redis
sudo cp utils/redis_init_script /etc/init.d/redis_6379
```

```
sudo vi /etc/init.d/redis_6379
```

Replace the beginning by:

(source: https://github.com/antirez/redis/issues/804#issuecomment-234132188)

#!/bin/sh # BEGIN INIT INFO # Provides: redis # Required-Start: $all # Required-Stop: $all # Default-Start: 2 3 4 5 # Default-Stop: 0 1 6 # Short-Description: starts the redis server # Description: starts redis using… # END INIT INFO

sudo cp redis.conf /etc/redis/6379.conf sudo mkdir /var/redis/6379

sudo vi /etc/redis/6379.conf

- Edit the configuration file, making sure to perform the following changes:
- Set daemonize to yes (by default it is set to no).
- Set the pidfile to /var/run/redis_6379.pid (modify the port if needed).
- Change the port accordingly. In our example it is not needed as the default port is already 6379.
- Set your preferred loglevel.
- Set the logfile to /var/log/redis_6379.log
- Set the dir to /var/redis/6379 (very important step!)

sudo update-rc.d redis_6379 defaults

Starting reddis:

sudo /etc/init.d/redis_6379 start

moving redis from one server to another

Do scp as described in ssh hell.

Make sure sure redis is stopped and appendonly mode is OFF in the config file.

Move dump.rdb to /var/lib/redis Change ownership and permission: chown redis:redis dump.rdb chmod 644 dump.rdb Start redis.

Change back appendonly to yes when redis has launched.

Install Elasticsearch

source: https://www.elastic.co/guide/en/elasticsearch/reference/current/deb.html

Download the public signing key:

```
wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -
```

Then:

```
sudo apt-get install apt-transport-https
```

```
echo "deb https://artifacts.elastic.co/packages/5.x/apt stable main" | sudo tee -a
/etc/apt/sources.list.d/elastic-5.x.list
```

```
sudo apt-get update
```

```
sudo apt-get install elasticsearch
```

```
sudo /bin/systemctl daemon-reload
```

```
sudo /bin/systemctl enable elasticsearch.service
```

Config

```
sudo vi /etc/elasticsearch/elasticsearch.yml
```

→ switch this param to true: bootstrap.memory_lock: true

You then need to make sure the JVM Heap size is no more than half the RAM. First fix a memory param:

```
sudo mkdir /etc/systemd/system/elasticsearch.service.d
cd  /etc/systemd/system/elasticsearch.service.d
sudo vi elasticsearch.conf
```

Add these lines:

LimitMEMLOCK=infinity

Adjust resource limits:

```
sudo vi /etc/security/limits.conf
```

Add line:

elasticsearch - nofile 65536

Add a jvm parameter:

sudo vi /etc/elasticsearch/jvm.options

Add this line:

-Djava.io.tmpdir=/var/tmp

Install the mongo to elasticsearch connection

elastic2-doc-manager

This is a doc manager by mongodb labs.

Source: https://github.com/mongodb-labs/elastic2-doc-manager

```
sudo apt-get install python-setuptools
sudo easy_install pip
sudo pip install 'elastic2-doc-manager[elastic5]'
sudo pip install 'mongo-connector[elastic5]'
```

run Mongo as a replicaset:

```
sudo service mongod stop
```

Create the path for your db (if needed)

sudo mkdir -p /data/db

```
sudo vi /etc/mongod.conf
```

```
Change dbPath to /data/db
```

Then:

```
sudo chown -R mongodb:mongodb /data/db
```

```
Then launch mongo as a replicaset:
```

```
sudo mongod --port 27017 --dbpath /data/db --replSet rs0 --fork --logpath
/var/log/mongodb.mongod.log
```

Install kibana

Kibana is the visualization engine for elastic.

```
sudo wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add
-
sudo apt-get install kibana
```

Configure Kibana to start automatically at boot:

```
sudo /bin/systemctl daemon-reload
sudo /bin/systemctl enable kibana.service
```

Install X-pack

```
https://www.elastic.co/guide/en/x-pack/current/installing-xpack.html
```

- it might need to create an empty file named /et/default/elasticsearch)
- see https://discuss.elastic.co/t/installing-x-pack-with-nonstandard-conf-dir/76448/3

  INFO

  > the second command (x-pack install for kibana) takes long minutes, that’s normal.

cd /usr/share/elasticsearch sudo bin/elasticsearch-plugin install x-pack

cd /usr/share/kibana sudo bin/kibana-plugin install x-pack

Disable the security component of X-Pack

This security component is hard to configure, and we don’t need it if we run elasticsearch behind a web server and a reverse proxy, on a single machine.

Add xpack.security.enabled: false

to /etc/elasticsearch/elasticsearch.yml

and to /etc/kibana/kibana.yml

Also in the same kibana.conf file, change the default username and passwd to "elastic" and "changeme" **and leave the quotes**

- start Elasticsearch: sudo /usr/share/elasticsearch/bin elasticsearch
- start Kibana: sudo /usr/share/kibana/bin kibana

Install the Mongo-connector for ElasticSearch:

Source: https://blog.jixee.me/how-to-use-mongo-connector-with-elasticsearch/

```
sudo apt-get install python2.7 python-pip curl
sudo pip install mongo-connector
```

Edit the conf of Mongo to turn on replicasets:

```
 sudo vi /etc/mongo.conf
(can also be: sudo vi /etc/mongod.conf)
```

Uncomment "replication", add two lines:

```
replication:
  replSetName: rs0
  oplogSizeMB: 100
```

```
sudo mongo-connector -m localhost:27017 -t localhost:9200 -d elastic2_doc_manager  -n
database1.collection1,database1.collection2
```

Start elasticsearch and Kibana

```
sudo service elasticsearch start
sudo systemctl start kibana.service
```

You can check that the connection is made here, your Mongo collections should be listed on this page:

```
http://localhost:9200/_cat/indices?v
```

the end

Author of this tutorial: Clement Levallois

All resources on linux security: https://seinecle.github.io/linux-security-tutorials/