**Redis**

Redis (расшифровывается как Remote Dictionary Server) – это быстрое хранилище данных типа «ключ-значение» в памяти с открытым исходным кодом для использования в качестве базы данных, кэша, брокера сообщений или очереди.

Настройка Redis

При установке Redis мы будем использовать внешний конфигурационный файл и внешний каталог хранения данных для того, чтобы при перезапуске контейнера вся информация оставалась в сохранности.

Нам понадобится два каталога — **/opt/redis/etc** и **/opt/redis/data**. В первом мы будем хранить настройки для **redis.conf**, а во втором — базы данных Redis и журналы, если вам требуется сохранять данные между перезапусками Redis. Обычно, если Redis используется для целей СУБД, а не кэша, то сохранение данных требуется.

Создадим каталоги:

sudo mkdir -p /opt/redis/{etc,data}

sudo mkdir /opt/redis/data/{bases,log}

Заполним конфигурационный файл **/opt/redis/etc/redis.conf** содержимым из блока, приведенного ниже:

bind 0.0.0.0

protected-mode yes

port 6379

tcp-backlog 511

timeout 0

tcp-keepalive 300

pidfile /var/run/redis/redis-server.pid

loglevel notice

logfile /data/log/redis-server.log

databases 16

always-show-logo yes

save 900 1

save 300 10

save 60 10000

stop-writes-on-bgsave-error yes

rdbcompression yes

rdbchecksum yes

dbfilename dump.rdb

dir /data/bases

slave-serve-stale-data yes

slave-read-only yes

repl-diskless-sync no

repl-diskless-sync-delay 5

repl-disable-tcp-nodelay no

slave-priority 100

lazyfree-lazy-eviction no

lazyfree-lazy-expire no

lazyfree-lazy-server-del no

slave-lazy-flush no

appendonly yes

appendfilename "appendonly.aof"

appendfsync everysec

no-appendfsync-on-rewrite no

auto-aof-rewrite-percentage 100

auto-aof-rewrite-min-size 64mb

aof-load-truncated yes

aof-use-rdb-preamble no

lua-time-limit 5000

slowlog-log-slower-than 10000

slowlog-max-len 128

latency-monitor-threshold 0

notify-keyspace-events ""

hash-max-ziplist-entries 512

hash-max-ziplist-value 64

list-max-ziplist-size -2

list-compress-depth 0

set-max-intset-entries 512

zset-max-ziplist-entries 128

zset-max-ziplist-value 64

hll-sparse-max-bytes 3000

activerehashing yes

client-output-buffer-limit normal 0 0 0

client-output-buffer-limit slave 256mb 64mb 60

client-output-buffer-limit pubsub 32mb 8mb 60

hz 10

aof-rewrite-incremental-fsync yes

Записи журнала Redis будут находиться в файле **/opt/redis/logs/redis-server.log**, а базы данных Redis в каталоге **/opt/redis/data/bases**.

Преимущества выполненных настроек Redis:

* все базы данных Redis будут сохраняться между перезагрузками контейнера;
* журналы redis будут находиться в файле, а не в syslog или в стандартном выводе контейнера;
* конфигурационный файл, использованный в настройке находится во внешнем файле и может быть изменен по необходимости.

Запустим контейнер:

sudo docker run --restart=always -d --name redis\_1 \

-v /opt/redis/etc/redis.conf:/usr/local/etc/redis/redis.conf \

-v /opt/redis/data:/data \

-p 127.0.0.1:6379:6379 redis redis-server \

/usr/local/etc/redis/redis.conf

Для проверки того, что контейнер нормально запустился запустите команду:

sudo docker ps

В результате, вы должны увидеть вывод, похожий на следующий:

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

e203d0bb6b92 redis "docker-entrypoint.s…" 18 seconds ago Up 17 seconds 127.0.0.1:6379->6379/tcp redis\_1

Простой запуск redis в docker контейнере:

docker run -d -p 6379:6379 redis

Ссылки:

1. <https://netpoint-dc.com/blog/nastroika-redis-docker-dlya-produktovogo-ispolzovaniya/>

# **Получить все данные из бд**

Далее переходим в терминал(консоль) и набираем команду:

redis-cli

Должно было установиться соединение с redis:

*127.0.0.1:6379>*  
  
Теперь для просмотра списка БД(ключей) нужно выполнить данную команду:

info keyspace

Получаем список БД(Ключей)

*db6:keys=77787,expires=77787,avg\_ttl=254468  
db9:keys=13,expires=12,avg\_ttl=86911  
db10:keys=28826,expires=28826,avg\_ttl=53236*

Перед нами 3 базы данных.   
Для перехода в интересующую БД необходимо выполнить команду:

select 6

Данная команда говорит нам о том, что мы переходим в базу данных с ключем 6.  
После выполнения команды, вы должны увидеть следующее:

*127.0.0.1:6379[6]>*

Теперь для получения списка всех данных из данной бд необходимо выполнить данную команду:

keys \*

После выполнения данной команды, вы должны увидеть список всех данных размещающихся в данной БД.  
Пример выполнения:

*1) "data1"  
2) "data2"  
3) "data3"  
4) "data4"  
5) "data5"*