Установка и настройка Redis для разных ОС

[Команда Timeweb Cloud](https://timeweb.cloud/tutorials/authors/timeweb-team)

[Команда Timeweb Cloud](https://timeweb.cloud/tutorials/authors/timeweb-team)

Наши инженеры, технические писатели, редакторы и маркетологи

30 ноября 2022 г.

5399

6 минут чтения

Поделиться

Redis — это система управления базами данных, которая хранит данные в формате «ключ-значение»: уникальному ключу в БД соответствует некоторое значение. Данные в этой базе данных хранятся в оперативной памяти, благодаря чему запросы обрабатываются быстро.

Согласно рейтингу [DB-Engines.com](https://db-engines.com/en/ranking), Redis — это самая популярная СУБД «ключ-значение». Благодаря этому типу, архитектура базы и запросы к ней проще, чем в реляционных баз данных. Однако использовать привычный SQL на стандартном Redis не получится: придется работать с Lua-скриптами или устанавливать модифицированные версии, например RediSQL.

Redis лучше всего использовать в проектах, где нужна база данных с быстрым доступом к данным и простой схемой.

Чтобы вы могли комфортно использовать эту СУБД в своих проектах, мы подготовили материал по установке Redis на три операционные системы: на Windows, Ubuntu и CentOS. По завершении установки мы настроим Redis, после чего СУБД будет готова к работе.

**Redis: установка на Windows**

Есть два варианта установки Redis на Windows:

* установка портированной версии;
* установка в WSL или Docker.

В этом разделе мы разберем установку портированной версии. Если вы хотите установить в WSL, то:

* руководство по установке и настройке WSL в Windows 10 вы найдете в материале «Как установить Node.js на Windows» в разделе «Установка WSL»;
* инструкция по установке на Ubuntu описана чуть ниже в разделе «Установка на Ubuntu».

Портированная версия для Windows вышла довольно давно: в 2016 году. Она размещена на github и доступна для скачивания всем пользователями. Преимущества установки этой версии заключается в её простоте: для установки не потребуется дополнительная «прослойка» в виде Docker или WSL. Выбирайте порт для Windows, если не хотите усложнять процесс установки и вас устраивает 6-летняя версия.

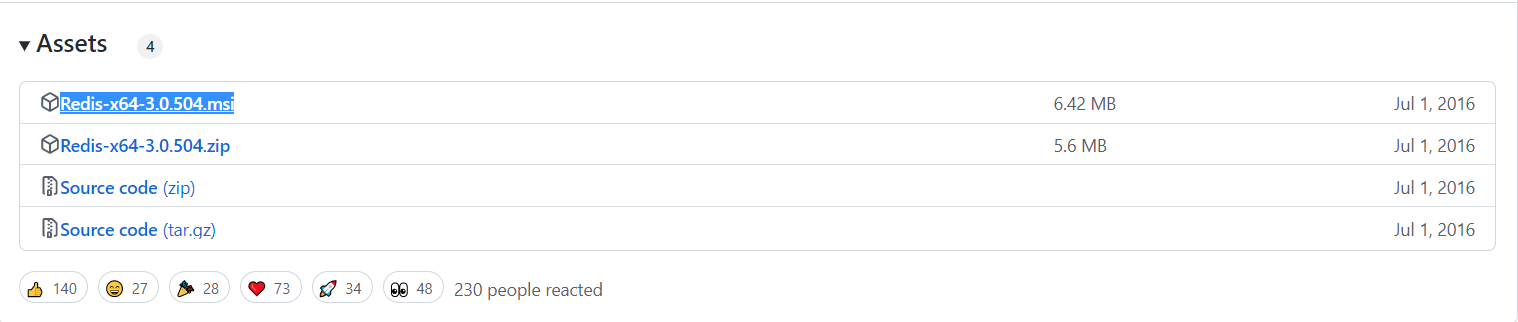
**Установка Redis на Windows**

Шаг 1 — заходим в репозиторий на GitHub портированной версии по ссылке: <https://github.com/microsoftarchive/redis>.

Шаг 2 — переходим на вкладку «Releases».

Шаг 3 — выбираем последний релиз.

Шаг 4 — скачиваем файл с расширением .msi. На изображении он выделен синим цветом:



Шаг 5 — после загрузки откройте файл. Начнется установка.

Шаг 6 — во время установки инсталлятор предложит вам добавить Redis в PATH. Если хотите использовать СУБД из командной строки, проставьте галочку напротив этого пункта.

Шаг 7 — во время выбора порта рекомендуется оставить стандартное значение 6379.

Шаг 8 — после установки необходимо перезагрузить компьютер.

После загрузки компьютера проверим работоспособность Redis. Для этого в командной строке выполняем команду **redis-server**.

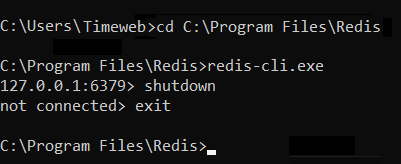
При её выполнении может возникнуть ошибка со следующим сообщением:  
Image7

Проблема заключается в том, что Windows в автоматическом режиме после установки регистрирует службу, к которой привязывает порт 6379. Когда выполняется команда redis-server, исполнитель обращается к файлу конфигурации и использует порт по-умолчанию, т.е. 6379. Из-за этого возникает ошибка.

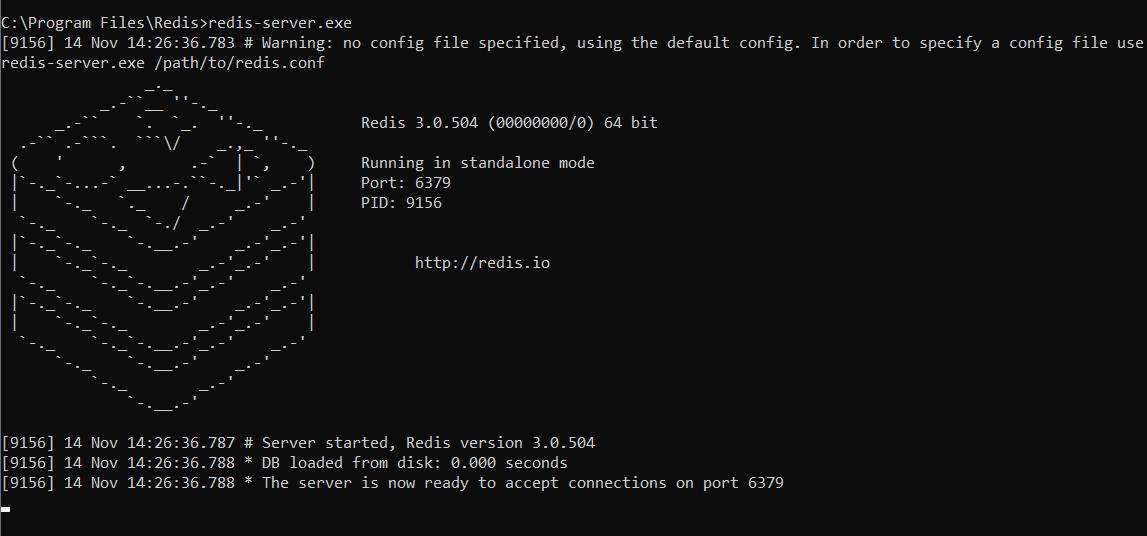
Чтобы её решить открываем командную строку и делаем следующее:

* переходим в каталог Redis командой cd;
* запускаем redis-cli.exe;
* выключаем сервер командой shutdown;
* выходим командой exit.

Вот набор команд для стандартного каталога:



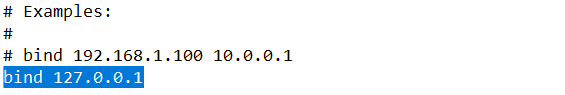
После этих действий команда redis-server успешно выполнилась:



Настройки Redis описываются в двух файлах: redis.windows.conf и redis.windows-service.conf.

1. Redis.windows-service.conf описывает настройки СУБД, запущенной в качестве службы. Этот формат подразумевает работу в фоновом режиме под управлением операционной системы (запуск при перезагрузке, перезапуск при сбоях и т.п).
2. Redis.windows.conf относится к использованию СУБД из командной строки с помощью redis-cli. Это же относится и к использованию СУБД в своих скриптах.

Настройка этих файлов приведет к изменению работы Redis. В нашем случае настроек минимальное количество: СУБД прослушивает любые подключения. Это не совсем безопасно, поэтому позволим подключение к Redis только с localhost. Для этого в файлах конфигурации находим задокументированную строку #bind 127.0.0.1 и удаляем решетку «#»:



**Установка Redis: Ubuntu 22.04**

Устанавливать Redis на Ubuntu мы будем из официального репозитория. Также, эта инструкция подойдет дляустановки Redis на Debian.

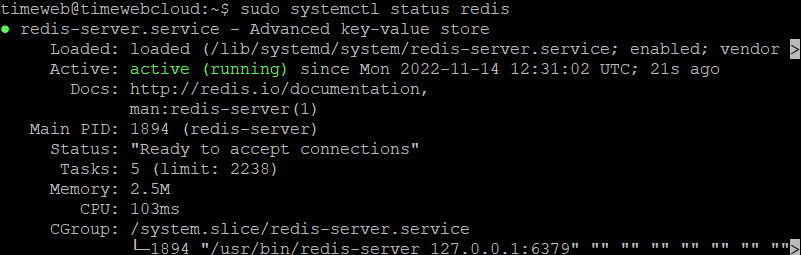
Шаг 1 — в первую очередь обновим индексы пакетов apt:

sudo apt update

Шаг 2 — загружаем Redis:

sudo apt install redis-server -y

Проверить работоспособность можно командой sudo systemctl status redis:



Служба активна.

**Установка Redis: CentOS 7**

Устанавливать Redis будем на CentOS 7. В первую очередь установим EPEL (Extra Packages for Enterprise Linux) — это пакет с дополнительными репозиториями, которые не включены в стандартную версию CentOS. Установим их:

sudo yum install epel-release

Теперь мы можем установить redis:

sudo yum install redis -y

Через несколько минут редис будет установлен на компьютер. Чтобы включить редис в автозагрузку, выполняем следующую команду:

sudo systemctl enable redis

**Настройка Redis**

Настройка Redis осуществляется через изменение конфигурационных файлов. Они для всех версий будут одинаковыми. Поэтому, для наглядной настройки, мы настроим Redis на операционной системе Ubuntu и удаленно подключимся к ней с Windows.

Для осуществления базовой настройки нам необходимо изменить 2 параметра: открыть удаленное подключение и установить пароль.

Шаг 1 — генерируем пароль:

openssl rand 25 | openssl base64 -A

Вывод:

/37DQhAt5MBq/34Lj24Ppn5LI/UZksAZJQ==

Злоумышленники могут перебирать до 150 тысяч паролей в секунду, поэтому важно выбрать надежный пароль. Копируем полученный результат и переходим к конфигурационному файлу.

Шаг 2 — открываем конфигурационный файл:

sudo nano /etc/redis/redis.conf

Шаг 3 — изменяем конфигурацию.

Найдем в конфигурационном файле строку «bind 127.0.0.1 ::1» и закомментируем её с помощью #. Не закрывая файл, найдем строку «protected-mode yes» и заменим её на «protected-mode no". В раздел «Security» после строки #requirepass foobared добавляем строку со сгенерированным паролем:

requirepass /37DQhAt5MBq/34Lj24Ppn5LI/UZksAZJQ==

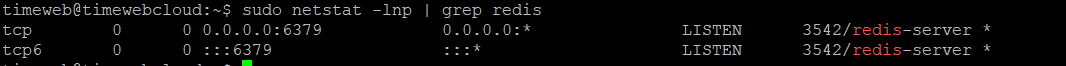
Закрываем файл и сохраняем все изменения.

Шаг 4 — перезапускаем Redis, чтобы изменения вступили в силу:

sudo systemctl restart redis.service

Шаг 5 — с помощью netstat проверяем, какие сетевые интерфейсы прослушивает Redis:

sudo netstat -lnp | grep redis



Шаг 6 — подключаемся удаленно. Попробуем подключиться к базе данных из командной строки Windows:

redis-cli -h 192.168.43.37 -p 6379 -a /37DQhAt5MBq/34Lj24Ppn5LI/UZksAZJQ==

Пропингуем СУБД:

192.168.43.37:6379> ping

PONG

И вставим кортеж:

192.168.43.37:6379> set key test

OK

192.168.43.37:6379> get key

"test"

**Redis как DBaaS**

DBaaS (database as a service) — это база данных, размещенная в облаке. Этот сервис похож на аренду сервера, только вместо виртуальной машины вы получаете базу данных. Такой вариант обладает рядом преимуществ:

* меньшее количество административных хлопот: не нужно обновлять СУБД и обслуживать оборудование;
* легкое масштабирование базы данных: если для вашего проекта потребовалась более производительная БД, то это легко сделать в панели управления;
* быстрый старт: СУБД будет установлена на старте;
* бесперебойная работа.

**Заключение**

В Timeweb.cloud есть услуга «Облачные базы данных» — уже готовое и настроенное решение. Если вы не хотите отягощать себя административными манипуляциями, то вы можете арендовать базу данных на Redis у Timeweb Cloud. Помимо Redis, Timeweb Cloud может предложить в качестве СУБД MySQL, PostgreSQL и MongoDB.