

# How to Start Kafka using Docker?

Start Kafka with Docker and run on all systems.

Running Kafka on Docker has the benefit that this method is going to work for all operating systems including Mac, Windows and Linux. It also helps to switch Kafka versions easily by simply updating the underlying docker-compose file.

## Pre-requisite: Install Docker & Docker Compose

### Install Docker

You must install Docker with the steps mentioned below to work.

Install Docker Desktop on [Mac](#).

Install Docker Desktop on [Windows](#).

Install Docker on [Linux](#) (choose your distro on the lefthand side menu).

### Installing Docker Compose

For Mac & Windows, if you have installed Docker Desktop, then Docker Compose is included as part of those desktop installs.

For Linux, [follow the steps here](#) (and do all the steps)

## How to run Kafka with Docker?

Docker Compose is a tool for defining and running multi-container Docker applications. As we'll start each of the Kafka services, e.g, zookeeper, broker, etc. in a different Docker container, docker-compose is a perfect tool to do that.

To use docker-compose, we have to define the services that make up our Kafka deployment in `docker-compose.yml` so they can be run together in an isolated environment.

Go to the GitHub project - <https://github.com/conduktor/kafka-stack-docker-compose> and clone or download the project. It has docker-compose files for many configurations.

```
1 $ git clone https://github.com/conduktor/kafka-stack-docker-compose.git
2 Cloning into 'kafka-stack-docker-compose'...
3 remote: Enumerating objects: 551, done.
4 remote: Counting objects: 100% (219/219), done.
5 remote: Compressing objects: 100% (125/125), done.
6 remote: Total 551 (delta 145), reused 140 (delta 83), pack-reused 332
7 Receiving objects: 100% (551/551), 110.29 KiB | 15.76 MiB/s, done.
8 Resolving deltas: 100% (375/375), done.
```

Next, go to the folder `kafka-stack-docker-compose` and list the files.

`.sh`

```
1 $ ls -l
2 total 60
3 -rw-rw-r-- 1 ec2-user ec2-user 4719 Nov 16 11:37 full-stack.yml
4 -rw-rw-r-- 1 ec2-user ec2-user 11345 Nov 16 11:37 LICENSE
5 -rw-rw-r-- 1 ec2-user ec2-user 8100 Nov 16 11:37 README.md
6 -rwxrwxr-x 1 ec2-user ec2-user 2150 Nov 16 11:37 test.sh
7 -rw-rw-r-- 1 ec2-user ec2-user 4759 Nov 16 11:37 zk-multiple-kafka-multiple-sche
8 -rw-rw-r-- 1 ec2-user ec2-user 4209 Nov 16 11:37 zk-multiple-kafka-multiple.yml
9 -rw-rw-r-- 1 ec2-user ec2-user 2728 Nov 16 11:37 zk-multiple-kafka-single.yml
10 -rw-rw-r-- 1 ec2-user ec2-user 2946 Nov 16 11:37 zk-single-kafka-multiple.yml
11 -rw-rw-r-- 1 ec2-user ec2-user 1604 Nov 16 11:37 zk-single-kafka-single.yml
```

Note the file `zk-single-kafka-single.yml`. As the name suggests, we'll use it to launch a Kafka cluster with a single Zookeeper and a single broker.

Now, use this command to launch a Kafka cluster with one Zookeeper and one Kafka broker. Add `-d` flag to run it in the background.

[Kafka Options Explorer](#)[Pricing](#)[Contact](#)[Start free](#)

Check to make sure both the services are running:

```
1 $ docker-compose -f zk-single-kafka-single.yml ps
2   Name                Command                  State              Ports
3   -----
4   kafka1              /etc/confluent/docker/run  Up                0.0.0.0:9092->9092/tcp, 0.0.0.0:9999-
5   zoo1                /etc/confluent/docker/run  Up                0.0.0.0:2181->2181/tcp, 2888/tcp, 3888/tcp
```

Kafka is going to be exposed on your computer at `localhost:9092`

## Running commands against our Kafka running on Docker

To run Kafka commands against Docker, we have two options:

Run commands directly from within the Docker container of Kafka (using `docker exec`)

Run commands from our host OS (we must first install the binaries)

### Option 1: Running commands from within the Kafka docker container

```
1 docker exec -it kafka1 /bin/bash
```

Then, from within the container, you can start running some Kafka commands (without `.sh`)

```
1 [appuser@kafka1 ~]$ kafka-topics --version
2 6.2.1-ccs (Commit:fa4bec046a2df3a6)
```

## Option 2: Running commands from outside your container

For this, you must install the Java and the Kafka Binaries on your system:

instructions for [Mac](#) (follow the whole document except starting Kafka and Zookeeper)

instructions for [Linux](#) (follow the whole document except starting Kafka and Zookeeper)

instructions for [Windows](#) (follow the whole document except starting Kafka and Zookeeper)

Once this is installed, the Kafka binaries should be in your path, and you will be good to issue your first Kafka command, for example `kafka-topics.sh`

## Stopping Kafka on Docker

Now that we've verified the installation, we can stop the services by stopping the respective containers:

```
1 $ docker-compose -f zk-single-kafka-single.yml stop
2 Stopping kafka1 ... done
3 Stopping zoo1 ... done
```

To remove all the resources including the containers altogether, use `down` instead of `stop`:

```
1 $ docker-compose -f zk-single-kafka-single.yml down
2 Removing kafka1 ... done
3 Removing zoo1 ... done
4 Removing network kafka-stack-docker-compose_default
```

Was this content helpful? 

 6

 0

PREVIOUS

NEXT

Share this page.



Apache, Apache Kafka, Kafka and the Kafka logo are trademarks of the **Apache Software Foundation**. All other trademarks, servicemarks, and copyrights are the property of their respective owners.

© 2023 Conduktor Inc. All Rights Reserved.

[Privacy Policy](#)[Terms of Service](#)[Cookie Policy](#)[DPA](#)[EULA](#)[Enterprise](#)

Status: **All Systems Operational**