

Configure and troubleshoot the Docker daemon

Estimated reading time: 11 minutes

After successfully installing and starting Docker, the `dockerd` daemon runs with its default configuration. This topic shows how to customize the configuration, start the daemon manually, and troubleshoot and debug the daemon if you run into issues.

Start the daemon using operating system utilities

On a typical installation the Docker daemon is started by a system utility, not manually by a user. This makes it easier to automatically start Docker when the machine reboots.

The command to start Docker depends on your operating system. Check the correct page under Install Docker (<https://docs.docker.com/install/>). To configure Docker to start automatically at system boot, see Configure Docker to start on boot (<https://docs.docker.com/install/linux/linux-postinstall/#configure-docker-to-start-on-boot>).

Start the daemon manually

If you don't want to use a system utility to manage the Docker daemon, or just want to test things out, you can manually run it using the `dockerd` command. You may need to use `sudo`, depending on your operating system configuration.

When you start Docker this way, it runs in the foreground and sends its logs directly to your terminal.

```
$ dockerd
```

```
INFO[0000] +job init_networkdriver()
INFO[0000] +job serveapi(unix:///var/run/docker.sock)
INFO[0000] Listening for HTTP on unix (/var/run/docker.sock)
```

To stop Docker when you have started it manually, issue a `Ctrl+C` in your terminal.

Configure the Docker daemon

There are two ways to configure the Docker daemon:

- Use a JSON configuration file. This is the preferred option, since it keeps all configurations in a single place.
- Use flags when starting `dockerd`.

