### What is Logging Drivers ?

Docker includes multiple logging mechanisms to help you [get information from running containers and services](https://docs.docker.com/config/containers/logging/). These mechanisms are called logging drivers. Each Docker daemon has a default logging driver, which each container uses unless you configure it to use a different logging driver, or “log-driver” for short.

There are lot some of them listed below

* + Json-file
  + None
  + Syslog
  + Local
  + Journald
  + Splunk
  + awslogs

**Json File Driver**

As a default, Docker uses the [json-file](https://docs.docker.com/config/containers/logging/json-file/)

By default, Docker captures the standard output (and standard error) of all your containers, and writes them in files using the JSON format. The JSON format annotates each line with its origin (stdout or stderr) and its timestamp. Each log file contains information about only one container.

{"log":"Log line is here\n","stream":"stdout","time":"2019-01-01T11:11:11.111111111Z"}

**Usage**

To use the json-file driver as the default logging driver, set the log-driver and log-opts keys to appropriate values in the daemon.json file, which is located in /etc/docker/ on Linux hosts or C:\ProgramData\docker\config\ on Windows Server. If the file does not exist, create it first

The following example sets the log driver to json-file and sets the max-size and max-file options to enable automatic log-rotation.

{

"log-driver": "json-file",

"log-opts": {

"max-size": "10m",

"max-file": "3"

}

}

Restart Docker for the changes to take effect for newly created containers

Existing containers do not use the new logging configuration

You can set the logging driver for a specific container by using the --log-driver flag to docker container create or docker run

$ docker run \

--log-driver json-file --log-opt max-size=10m \

alpine echo hello world

Example

This example starts an alpine container which can have a maximum of 3 log files no larger than 10 megabytes each.

$ docker run -it --log-opt max-size=10m --log-opt max-file=3 alpine ash

**AWS CloudWatch Logs Logging Driver**

The awslogs logging driver sends container logs to [Amazon CloudWatch Logs](https://aws.amazon.com/cloudwatch/details/#log-monitoring). Log entries can be retrieved through the [AWS Management Console](https://console.aws.amazon.com/cloudwatch/home#logs:) or the [AWS SDKs and Command Line Tools](https://docs.aws.amazon.com/cli/latest/reference/logs/index.html).

Modify daemon.json and restart it

{

"log-driver": "awslogs",

"log-opts": {

"awslogs-region": "us-east-1"

}

}

$ docker run --log-driver=awslogs ...

### Docker logs

Fetch the logs of a container

$ docker logs [OPTIONS] CONTAINER

* --details Show extra details provided to logs
* --follow , -f Follow log output
* --since Show logs since timestamp (e.g. 2013-01-02T13:23:37Z) or relative (e.g. 42m for 42 minutes)
* --tail , -n all Number of lines to show from the end of the logs
* --timestamps , -t Show timestamps
* --until Show logs before a timestamp (e.g. 2013-01-02T13:23:37Z) or relative (e.g. 42m for 42 minutes)

**Example**

* The docker logs --follow command will continue streaming the new output from the container’s STDOUT and STDERR.
* Passing a negative number or a non-integer to --tail is invalid and the value is set to all in that case.
* The docker logs --details command will add on extra attributes, such as environment variables and labels, provided to --log-opt when creating the container.

In order to retrieve logs before a specific point in time, run:

$ docker logs -f --until=2s test