JSON File logging driver

- Usage
  - Options
  - Examples

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"
}
```

> [!WARNING]
>
> The `json-file` logging driver uses file-based storage. These files are designed to be exclusively accessed by the Docker daemon. Interacting with these files with external tools may interfere with Docker's logging system and result in unexpected behavior, and should be avoided.

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. For more information about configuring Docker using `daemon.json` , see daemon.json.

{{% include "daemon-cfg-desktop.md" %}}

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"
  }
}
```

> [!NOTE]
>
> `log-opts` configuration options in the `daemon.json` configuration file must be provided as strings. Boolean and numeric values (such as the value for `max-file` in the example above) must therefore be enclosed in quotes (`"`).

Restart Docker for the changes to take effect for newly created containers. Existing containers don't use the new logging configuration automatically.

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
```

Options

The `json-file` logging driver supports the following logging options:

Option	Description	Example value
`max-size`	The maximum size of the log before it is rolled. A positive integer plus a unit modifier (`k`, `m`, or `g`; case-insensitive). Defaults to -1 (unlimited).	`--log-opt max-size=10m`
`max-file`	The maximum number of log files that can be present. If rolling the logs creates excess files, the oldest file is removed. **Only effective when `max-size` is also set.** A positive integer. Defaults to 1.	`--log-opt max-file=3`

Option	Description	Example value
`labels`	Applies when starting the Docker daemon. A comma-separated list of logging-related labels this daemon accepts. Used for advanced log tag options.	`--log-opt labels=production_status,geo`
`labels-regex`	Similar to and compatible with `labels`. A regular expression to match logging-related labels. Used for advanced log tag options.	`--log-opt labels-regex=^(production_status|geo)`
`env`	Applies when starting the Docker daemon. A comma-separated list of logging-related environment variables this daemon accepts. Used for advanced log tag options.	`--log-opt env=os,customer`
`env-regex`	Similar to and compatible with `env`. A regular expression to match logging-related environment variables. Used for advanced log tag options.	`--log-opt env-regex=^(os|customer)`
`compress`	Toggles compression for rotated logs. Defaults to `false` (no compression).	`--log-opt compress=true`

Examples

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
```

□□□ #2
□ □□□ □□□ 2026-10-31 19:24:51 UTC
□ □□□ □□□ 2026-10-31 19:24:53 UTC