

Journald logging driver

Estimated reading time: 5 minutes

The `journald` logging driver sends container logs to the `systemd` journal (<http://www.freedesktop.org/software/systemd/man/systemd-journald.service.html>). Log entries can be retrieved using the `journalctl` command, through use of the `journal` API, or using the `docker logs` command.

In addition to the text of the log message itself, the `journald` log driver stores the following metadata in the journal with each message:

Field	Description
<code>CONTAINER_ID</code>	The container ID truncated to 12 characters.
<code>CONTAINER_ID_FULL</code>	The full 64-character container ID.
<code>CONTAINER_NAME</code>	The container name at the time it was started. If you use <code>docker rename</code> to rename a container, the new name is not reflected in the journal entries.
<code>CONTAINER_TAG</code> , <code>SYSLOG_IDENTIFIER</code>	The container tag (log tag option documentation (https://docs.docker.com/config/containers/logging/log_tags/)).
<code>CONTAINER_PARTIAL_MESSAGE</code>	A field that flags log integrity. Improve logging of long log lines.

Usage

To use the `journald` driver as the default logging driver, set the `log-driver` and `log-opt` keys to appropriate values in the `daemon.json` file, which is located in `/etc/docker/` on Linux hosts or `C:\ProgramData\docker\config\daemon.json` on Windows Server. For more about configuring Docker using `daemon.json`, see `daemon.json` (<https://docs.docker.com/engine/reference/commandline/dockerd/#daemon-configuration-file>).

The following example sets the log driver to `journald`:

```
{
  "log-driver": "journald"
}
```

Restart Docker for the changes to take effect.

To configure the logging driver for a specific container, use the `--log-driver` flag on the `docker run` command.

```
$ docker run --log-driver=journald ...
```

Options

Use the `--log-opt NAME=VALUE` flag to specify additional `journald` logging driver options.

Option	Required	Description
<code>tag</code>	optional	Specify template to set <code>CONTAINER_TAG</code> and <code>SYSLOG_IDENTIFIER</code> value in <code>journald</code> logs. Refer to log tag option documentation (https://docs.docker.com/engine/admin/logging/log_tags/) to customize the log tag format
<code>labels</code>	optional	Comma-separated list of keys of labels, which should be included in message, if these labels are specified for the container.
<code>env</code>	optional	Comma-separated list of keys of environment variables, which should be included in message, if these variables are specified for the container.
<code>env-regex</code>	optional	Similar to and compatible with <code>env</code> . A regular expression to match logging-related environment variables. Used for advanced log tag options (https://docs.docker.com/engine/admin/logging/log_tags/).

If a collision occurs between label and `env` keys, the value of the `env` takes precedence. Each option adds additional fields to the attributes of a logging message.

Below is an example of the logging options required to log to `journald`.

```
$ docker run --log-driver=journald \  
  --log-opt labels=location \  
  --log-opt env=TEST \  
  --env "TEST=false" \  
  --label location=west \  
  your/application
```

This configuration also directs the driver to include in the payload the label `location`, and the environment variable `TEST`. If the `--env "TEST=false"` or `--label location=west` arguments were omitted, the corresponding key would not be set in the `journald` log.

Note regarding container names

The value logged in the `CONTAINER_NAME` field is the name of the container that was set at startup. If you use `docker rename` to rename a container, the new name **is not reflected** in the journal entries. Journal entries continue to use the original name.

Retrieve log messages with `journalctl`

Use the `journalctl` command to retrieve log messages. You can apply filter expressions to limit the retrieved messages to those associated with a specific container:

```
$ sudo journalctl CONTAINER_NAME=webserver
```

You can use additional filters to further limit the messages retrieved. The `-b` flag only retrieves messages generated since the last system boot:

```
$ sudo journalctl -b CONTAINER_NAME=webserver
```

The `-o` flag specifies the format for the retrieved log messages. Use `-o json` to return the log messages in JSON format.

```
$ sudo journalctl -o json CONTAINER_NAME=webserver
```

View logs for a container with a TTY enabled

If TTY is enabled on a container you may see `[10B blob data]` in the output when retrieving log messages. The reason for that is that `\r` is appended to the end of the line and `journalctl` doesn't strip it automatically unless `--all` is set:

```
$ sudo journalctl -b CONTAINER_NAME=webserver --all
```

Retrieve log messages with the `journal` API

This example uses the `systemd` Python module to retrieve container logs:

```
import systemd.journal

reader = systemd.journal.Reader()
reader.add_match('CONTAINER_NAME=web')

for msg in reader:
    print '{CONTAINER_ID_FULL}: {MESSAGE}'.format(**msg)
```

Journald (<https://docs.docker.com/glossary/?term=Journald>), docker (<https://docs.docker.com/glossary/?term=docker>), logging (<https://docs.docker.com/glossary/?term=logging>), driver (<https://docs.docker.com/glossary/?term=driver>)