<epam>

Using journalctl

CentOS Linux Essentials

TRAINING

CENTER

<epam>

<epam>

Agenda

- what is journalctl
- basic log levels
- watching logs of particular unit
- watching errors
- checking logs with particular tag
- watching logs for particular period

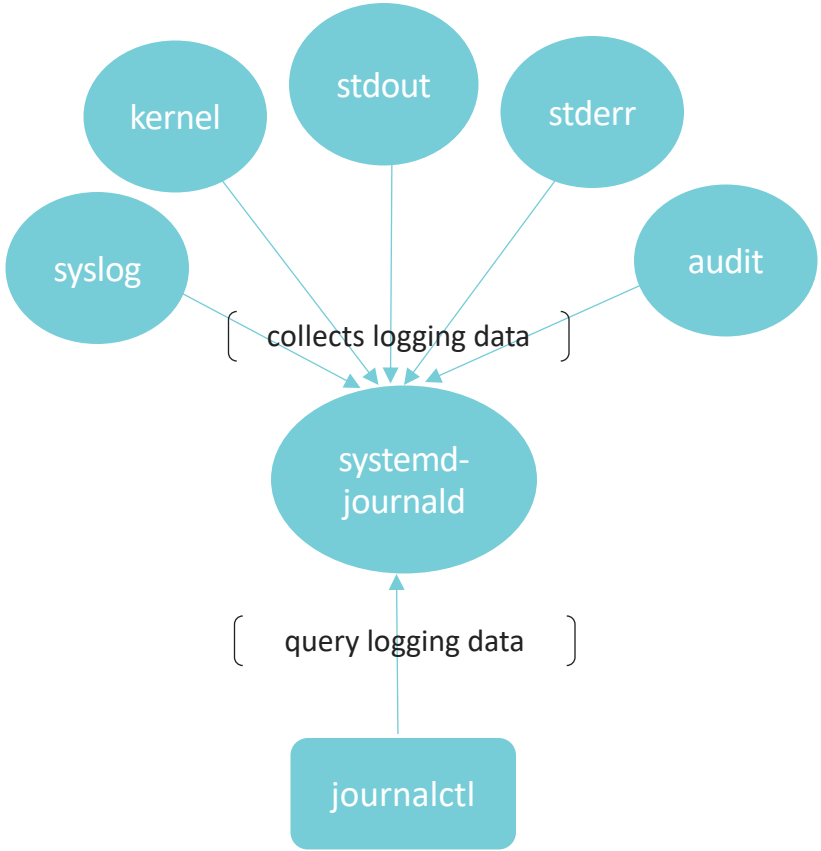
What is journalctl?

In modern Linux systems logging subsystem is implemented by **systemd-journald.** It's a system service that collects and stores logging data.

It creates and maintains structured, indexed binary files called **journals** based on logging information that is received from a variety of sources.

One of the impetuses behind the systemd journal is to **centralize** the management of logs regardless of where the messages are originating.

journalctl is a utility to query and display the systemd journal data.



Basic log levels

DEBUG	• debug-level **messages**
INFO	• Informational
NOTICE	• **normal but significant condition**
WARNING	• warning **conditions**
ERR	• **error conditions**
CRIT	• **critical conditions**
ALERT	• action **must be taken immediately**
EMERG	• **system is unusable**

Watching logs of particular unit

To see messages logged by any systemd unit, use the **-u** switch.

```
$ journalctl -u nginx.service
```

The **-u** switch can be used multiple times to specify more than one unit source.

```
$ journalctl -u nginx.service -u mysql.service
```

Watching errors

To show only entries logged at the **error** level or above, you can type:

```
$ journalctl -p err -b -x
```

where:

- **-p** means **priority** or **log level**
- **-b** means since last boot
- **-x** means to add explanatory help texts to log messages in the output where this is available. These short help texts will explain the context of an error or log event, possible solutions, as well as pointers to support forums, developer documentation, and any other relevant manuals

Checking logs with particular tag

To show only entries for specified syslog identifier:

```
$ journalctl -t dockerd
```

-- Logs begin at Fri 2020-03-13 13:17:01 +03, end at Sun 2020-04-26 11:40:10 +03. --
Mar 16 14:23:56 localhost dockerd[2009]: time="2020-03-16T14:23:56" level=info msg="ignoring event" modu...
Mar 16 14:25:04 localhost dockerd[2009]: time="2020-03-16T14:25:04" level=info msg="ignoring event" modu...
Mar 16 14:31:59 localhost dockerd[2009]: time="2020-03-16T14:31:59" level=info msg="parsed scheme: \"\"...
Mar 16 14:31:59 localhost dockerd[2009]: time="2020-03-16T14:31:59" level=info msg="scheme \"\" not regi...
Mar 16 14:31:59 localhost dockerd[2009]: time="2020-03-16T14:31:59" level=info msg="ccResolverWrapper: s...
Mar 16 14:31:59 localhost dockerd[2009]: time="2020-03-16T14:31:59" level=info msg="ClientConn switching...
Mar 16 14:32:26 localhost dockerd[2009]: time="2020-03-16T14:32:26" level=info msg="Layer sha256:886f8f9...
Mar 16 14:35:13 localhost dockerd[2009]: time="2020-03-16T14:35:13" level=info msg="ignoring event" modu...
Mar 16 14:35:25 localhost dockerd[2009]: time="2020-03-16T14:35:25" level=info msg="ignoring event" modu...
Mar 16 14:35:26 localhost dockerd[2009]: time="2020-03-16T14:35:26" level=info msg="Layer sha256:c0c0d46...
Mar 16 14:35:26 localhost dockerd[2009]: time="2020-03-16T14:35:26" level=info msg="Layer sha256:c0c0d46...
Mar 16 14:35:26 localhost dockerd[2009]: time="2020-03-16T14:35:26" level=info msg="Layer sha256:c0c0d46...

Watching logs for particular period

To see messages logged within a specific time window, we can use the **--since** and **--until** options.

```
$ journalctl --since "1 hour ago"
```

```
$ journalctl --since "2 days ago"
```

```
$ journalctl --since "2015-06-26 23:15:00" --until "2015-06-26 23:20:00"
```