**RHEL7: Locate and interpret system log files and journals.**

**Presentation**

Most of system log files are located in the **/var/log** directory due to **SYSLOG** default configuration (see **/etc/rsyslog.conf** file).

In addition, all **SELinux** events are written into the **/var/log/audit/audit.log** file.

With **Systemd**, new commands have been created to analyse logs at boot time and later.

**Boot Process**

**Systemd** primary task is to manage the boot process and provides informations about it.  
To get the boot process duration, type:

# **systemd-analyze**

Startup finished in 422ms (kernel) + 2.722s (initrd) + 9.674s (userspace) = 12.820s

To get the time spent by each task during the boot process, type:

# **systemd-analyze blame**

7.029s network.service

2.241s plymouth-start.service

1.293s kdump.service

1.156s plymouth-quit-wait.service

1.048s firewalld.service

632ms postfix.service

621ms tuned.service

460ms iprupdate.service

446ms iprinit.service

344ms accounts-daemon.service

...

7ms systemd-update-utmp-runlevel.service

5ms systemd-random-seed.service

5ms sys-kernel-config.mount

Note: You will find additional information on this point in the [Lennart Poettering’s blog](http://0pointer.de/blog/projects/blame-game.html).

**Journal Analysis**

In addition, **Systemd** handles the system event log, a **syslog** daemon is not mandatory any more.  
The reasons behind **Journald** creation are explained in this [Introduction to Journald](https://docs.google.com/document/pub?id=1IC9yOXj7j6cdLLxWEBAGRL6wl97tFxgjLUEHIX3MSTs&pli=1).

To get the content of the **Systemd** journal, type:

# **journalctl**

To get all the events related to the **crond** process in the journal, type:

# **journalctl /sbin/crond**

Note: You can replace **/sbin/crond** by **`which crond`**.

**Altenatively**, to get all the events related to the **crond** process, you can also type:

# **journalctl -u crond**

To get all the events since the last boot, type:

# **journalctl -b**

To get all the events that appeared today in the journal, type:

# **journalctl --since=today**

To get all the events with a syslog priority of err, type:

# **journalctl -p err**

To get the 10 last events and wait for any new one (like **tail -f /var/log/messages**), type:

# **journalctl -f**

**Permanent Storage**

By default, **Journald** logs are stored in the **/run/log/journal** directory and disappear after a reboot.  
To store **Journald** logs in a more permanent way, type:

# **mkdir /var/log/journal**

# **echo "SystemMaxUse=50M" >> /etc/systemd/journald.conf**

# **systemctl restart systemd-journald**

Note: Setting the **SystemMaxUse** variable is necessary because otherwise **10%** of the filesystem where the **/var/log/journal** directory is stored may be used at maximum by the journal.

* On a RHEL 7 machine, the logger command does not generate a new log entry in /var/log/messages -- instead, it's empty except for messages from rsyslogd itself, e.g.:
* **~]# cat /var/log/messages**
* Jan 26 11:32:45 r72 rsyslogd: [origin software="rsyslogd" swVersion="7.4.7" x-pid="1704" x-info="http://www.rsyslog.com"] rsyslogd was HUPed
* **~]# systemctl restart rsyslog**
* **~]# logger testing testing**
* **~]# cat /var/log/messages**
* Jan 26 11:32:45 r72 rsyslogd: [origin software="rsyslogd" swVersion="7.4.7" x-pid="1704" x-info="http://www.rsyslog.com"] rsyslogd was HUPed
* Jan 26 11:33:10 r72 rsyslogd: [origin software="rsyslogd" swVersion="7.4.7" x-pid="1704" x-info="http://www.rsyslog.com"] exiting on signal 15.
* Jan 26 11:33:10 r72 rsyslogd: [origin software="rsyslogd" swVersion="7.4.7" x-pid="1792" x-info="http://www.rsyslog.com"] start
* On a RHEL 7 machine, the logger command generates new log entries in the systemd journal ...
* **~]# journalctl -fn0 &**
* [1] 1884
* -- Logs begin at Tue 2016-01-26 10:26:17 EST. --
* **~]# logger testing logger**
* Jan 26 11:38:42 r72.example.com root[1885]: testing logger

However, the same messages are not showing up in rsyslog's /var/log/messages file.

**~]# tail -fn0 /var/log/messages &**

[1] 1913

**~]# logger testing logger again**

**~]#**