

# How to Use journalctl to Read Linux System Logs

[Dave McKay @TheGurkha](#)

February 3, 2020, 8:00am EDT



[Fatmawati Achmad Zaenuri/Shutterstock](#)

Linux system logging changed with the introduction of `systemd`. Learn how to use the `journalctl` command to read and filter system log messages.

## Centralized Logging

No stranger to controversy, the [systemd system and service manager](#) introduced a significant change in the way system logs are gathered. Logs used to be located at different places in the file system according to the service or daemon that was creating them. But they all had one thing in common. They were plain text files.

With `systemd` all the system, boot, and kernel log files are collected and managed by a central, dedicated logging solution. The format they are stored in is a binary one. One thing this facilitates is being able to extract the data in different formats, such as [JSON](#), as we shall see.

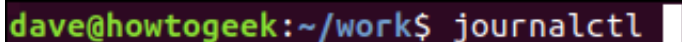
it can also make it easier to cross-reference related information that would have previously been recorded in separate log files. Because the data is now held in a single journal, the data from several sources of interest can be selected and displayed in a single interwoven list of entries.

`journalctl` is the tool [used to work with the journal](#).

## journalctl With No Frills

You can invoke `journalctl` with no command line parameters:

```
journalctl
```

A terminal window with a dark purple background. The prompt is 'dave@howtogeek:~/work\$' in green and blue. The command 'journalctl' is entered in white, followed by a white cursor block.

```
dave@howtogeek:~/work$ journalctl
```

`journalctl` displays the entire journal, with the oldest entries at the top of the list. The list is displayed in `less`, allowing you to page and search using the usual navigation features of `less`. You can also use the `Left Arrow` and `Right Arrow` keys to scroll sideways to read wide log entries.

```
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 07
Sep 30 14:55:07 howtogeek kernel: Linux version 5.0.0-29-generic (buil
Sep 30 14:55:07 howtogeek kernel: Command line: BOOT_IMAGE=/boot/vmlin
Sep 30 14:55:07 howtogeek kernel: KERNEL supported cpus:
Sep 30 14:55:07 howtogeek kernel: Intel GenuineIntel
Sep 30 14:55:07 howtogeek kernel: AMD AuthenticAMD
Sep 30 14:55:07 howtogeek kernel: Hygon HygonGenuine
Sep 30 14:55:07 howtogeek kernel: Centaur CentaurHauls
Sep 30 14:55:07 howtogeek kernel: [Firmware Bug]: TSC doesn't count wi
Sep 30 14:55:07 howtogeek kernel: x86/fpu: Supporting XSAVE feature 0x
Sep 30 14:55:07 howtogeek kernel: x86/fpu: Supporting XSAVE feature 0x
Sep 30 14:55:07 howtogeek kernel: x86/fpu: Supporting XSAVE feature 0x
Sep 30 14:55:07 howtogeek kernel: x86/fpu: xstate_offset[2]: 576, xst
Sep 30 14:55:07 howtogeek kernel: x86/fpu: Enabled xstate features 0x7
Sep 30 14:55:07 howtogeek kernel: BIOS-provided physical RAM map:
Sep 30 14:55:07 howtogeek kernel: BIOS-e820: [mem 0x0000000000000000-0
Sep 30 14:55:07 howtogeek kernel: BIOS-e820: [mem 0x0000000000009fc0-0
Sep 30 14:55:07 howtogeek kernel: BIOS-e820: [mem 0x000000000000f000-0
Sep 30 14:55:07 howtogeek kernel: BIOS-e820: [mem 0x0000000000100000-0
Sep 30 14:55:07 howtogeek kernel: BIOS-e820: [mem 0x0000000007fff0000-0
lines 1-20
```

Pressing the `End` key will hop straight to the bottom of the list, and the newest log entries.

```
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: Parsed $name: /home/d
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: New uri after exists
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: Trying to save file t
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: Saving file /home/dav
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: Updating fileinfos fo
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: file:///home/dave/Pic
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: file:///home/dave/Pic
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: screenshot successful
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: fct_update_tray_menu
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: section was emitted b
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: 2 Ns used in wild-car
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: Searching for files w
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: Comparing journalctl_
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: 01 is currently great
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: dave@howtogeek: ~/wor
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: Type: motion-notify
Jan 12 07:04:08 howtogeek shutter.desktop[9975]: Type: motion-notify
Jan 12 07:04:10 howtogeek shutter.desktop[9975]: Type: motion-notifGLi
Jan 12 07:04:10 howtogeek shutter.desktop[9975]: GLib-CRITICAL **: Sou
Jan 12 07:04:10 howtogeek org.gnome.Shell.desktop[1698]: Window manage
lines 60534-60553/60553 (END)
```

Press `ctrl+c` to exit.

## RELATED: [How to Use the less Command on Linux](#)

Although `journalctl` can be called without using `sudo`, [you will ensure you see all the detail](#) within the log if you do use `sudo`.

```
sudo journalctl
```

```
dave@howtogeek:~/work$ sudo journalctl
```

If you need to, you can make `journalctl` send its output to the terminal window instead of to `less`, by using the `--no-pager` option.

```
sudo journalctl --no-pager
```

```
dave@howtogeek:~/work$ sudo journalctl --no-pager
```

The output scrolls quickly through the terminal window, and you are returned to the command prompt.

```
est value...
Jan 12 07:07:20 howtogeek shutter.desktop[9975]: ComparinINFO: DBus co
nnection to org.freedesktop.compiz failed --> skipping compiz related
tasks
Jan 12 07:07:20 howtogeek shutter.desktop[9975]: org.freedesktop.DBus.
Error.ServiceUnknown: The name org.freedesktop.compiz was not provided
by any .service files
Jan 12 07:07:21 howtogeek shutter.desktop[9975]: GLib-CRITICAL **: Sou
rce ID 1862 was not found when attempting to remove it at /usr/bin/shu
tter line 7262.
Jan 12 07:07:21 howtogeek shutter.desktop[9975]: GLib-CRITICAL **: Sou
rce ID 1864 was not found when attempting to remove it at /usr/share/p
erl5/Shutter/App/ShutterNotification.pm line 186.
Jan 12 07:07:21 howtogeek org.gnome.Shell.desktop[1698]: Window manage
r warning: Buggy client sent a _NET_ACTIVE_WINDOW message with a times
tamp of 0 for 0x280007c (/home/dave)
Jan 12 07:07:24 howtogeek sudo[10124]:      dave : TTY=pts/0 ; PWD=/hom
e/dave/work ; USER=root ; COMMAND=/bin/journalctl --no-pager
Jan 12 07:07:24 howtogeek sudo[10124]: pam_unix(sudo:session): session
opened for user root by (uid=0)
dave@howtogeek:~/work$
```

To limit the number of lines that `journalctl` returns, use the `-n` (lines) option. Let's ask for ten lines of output:

```
sudo journalctl -n 10
```



```
dave@howtogeek:~/work$ sudo journalctl -n 10
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 07
Jan 12 07:07:27 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:07:27 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:07:27 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:07:27 howtogeek shutter.desktop[9975]: INFO: DBus connection
Jan 12 07:07:27 howtogeek shutter.desktop[9975]: org.freedesktop.DBus.
Jan 12 07:07:29 howtogeek shutter.desktop[9975]: GLib-CRITICAL **: Sou
Jan 12 07:07:29 howtogeek shutter.desktop[9975]: GLib-CRITICAL **: Sou
Jan 12 07:07:29 howtogeek org.gnome.Shell.desktop[1698]: Window manage
Jan 12 07:07:48 howtogeek sudo[10142]:      dave : TTY=pts/0 ; PWD=/hom
Jan 12 07:07:48 howtogeek sudo[10142]: pam_unix(sudo:session): session
lines 1-11/11 (END)
```

## Following Journal Updates

To make `journalctl` display the newest entries as they arrive in the journal, use the `-f` (follow) option.

```
sudo journalctl -f
```

```
dave@howtogeek:~/work$ sudo journalctl -f
```

```
Dimensions: 16 x 16 at /usr/bin/shutter line 2891.
Jan 12 07:09:07 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Rowstride: 64, Length: 1048 at /usr/bin/shutter line 2891.
Jan 12 07:09:07 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Copy pixels == false at /usr/bin/shutter line 2891.
Jan 12 07:09:07 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **: gdk
_pixbuf_from_pixdata() called on: at /usr/bin/shutter line 2891.
Jan 12 07:09:07 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Encoding raw at /usr/bin/shutter line 2891.
Jan 12 07:09:07 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Dimensions: 16 x 16 at /usr/bin/shutter line 2891.
Jan 12 07:09:07 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Rowstride: 64, Length: 1048 at /usr/bin/shutter line 2891.
Jan 12 07:09:07 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Copy pixels == false at /usr/bin/shutter line 2891.
Jan 12 07:09:07 howtogeek shutter.desktop[9975]: INFO: Dbus connection
to org.freedesktop.compiz failed --> skipping compiz related tasks
Jan 12 07:09:07 howtogeek shutter.desktop[9975]: org.freedesktop.DBus.
Error.ServiceUnknown: The name org.freedesktop.compiz was not provided
by any .service files
```

The newest entry has a timestamp of 07:09:07. As new activity takes place, the new entries are appended to the bottom of the display. Near real-time updates—cool!

```
Jan 12 07:09:07 howtogeek shutter.desktop[9975]: org.freedesktop.DBus.  
Error.ServiceUnknown: The name org.freedesktop.compiz was not provided  
by any .service files  
Jan 12 07:09:09 howtogeek shutter.desktop[9975]: GLib-CRITICAL **: Sou  
rce ID 2610 was not found when attempting to remove it at /usr/bin/shu  
tter line 7262.  
Jan 12 07:09:09 howtogeek shutter.desktop[9975]: GLib-CRITICAL **: Sou  
rce ID 2612 was not found when attempting to remove it at /usr/share/p  
erl5/Shutter/App/ShutterNotification.pm line 186.  
Jan 12 07:09:09 howtogeek org.gnome.Shell.desktop[1698]: Window manage  
r warning: Buggy client sent a _NET_ACTIVE_WINDOW message with a times  
tamp of 0 for 0x280007c (/home/dave)  
Jan 12 07:09:58 howtogeek geek-app[10188]: New Message from HTG  
Jan 12 07:10:03 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **: gdk  
_pixbuf_from_pixdata() called on: at /usr/bin/shutter line 2891.  
Jan 12 07:10:03 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:   
Encoding raw at /usr/bin/shutter line 2891.  
Jan 12 07:10:03 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:   
Dimensions: 16 x 16 at /usr/bin/shutter line 2891.  
Jan 12 07:10:03 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:   
Rowstride: 64, Length: 1048 at /usr/bin/shutter line 2891.
```

At 07:09:59 an application called `geek-app` injected a log entry into the journal that said, "New Message from HTG."

## Changing the Display Format

Because the journal is a binary file, the data in it needs to be translated or parsed into text before it can be displayed to you. With different parsers, different output formats can be created from the same binary source data. There are several different formats that `journalctl` can use.

The default output is the short format, which is very similar to the classic system log format. To explicitly request the short format, use the `-o` (output) option with the `short` modifier.

```
sudo journalctl -n 10 -o short-full
```



```
dave@howtogeek:~/work$ sudo journalctl -n 10 -o short
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 07
Jan 12 07:11:15 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:11:15 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:11:15 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:11:15 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:11:47 howtogeek sudo[10164]: pam_unix(sudo:session): session
Jan 12 07:17:01 howtogeek CRON[10217]: pam_unix(cron:session): session
Jan 12 07:17:01 howtogeek CRON[10218]: (root) CMD ( cd / && run-part
Jan 12 07:17:01 howtogeek CRON[10217]: pam_unix(cron:session): session
Jan 12 07:18:12 howtogeek sudo[10220]: dave : TTY=pts/0 ; PWD=/hom
Jan 12 07:18:12 howtogeek sudo[10220]: pam_unix(sudo:session): session
lines 1-11/11 (END)
```

From left to right, the fields are:

- The time the message was created, in local time.
- The hostname.
- The process name. This is the process that generated the message.
- The log message.

To obtain a complete date and time stamp use the `short-full` modifier:

```
sudo journalctl -n 10 -o short-full
```

```
dave@howtogeek:~/work$ sudo journalctl -n 10 -o short-full
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 07:18:58 EST.
Sun 2020-01-12 07:18:22 EST howtogeek shutter.desktop[9975]: org.freedesktop
Sun 2020-01-12 07:18:23 EST howtogeek shutter.desktop[9975]: GLib-CRIT
Sun 2020-01-12 07:18:23 EST howtogeek shutter.desktop[9975]: GLib-CRIT
Sun 2020-01-12 07:18:23 EST howtogeek org.gnome.Shell.desktop[1698]: W
Sun 2020-01-12 07:18:27 EST howtogeek sudo[10220]: pam_unix(sudo:sessi
Sun 2020-01-12 07:18:36 EST howtogeek sudo[10225]:      dave : TTY=pts/
Sun 2020-01-12 07:18:36 EST howtogeek sudo[10225]: pam_unix(sudo:sessi
Sun 2020-01-12 07:18:55 EST howtogeek sudo[10225]: pam_unix(sudo:sessi
Sun 2020-01-12 07:18:58 EST howtogeek sudo[10228]:      dave : TTY=pts/
Sun 2020-01-12 07:18:58 EST howtogeek sudo[10228]: pam_unix(sudo:sessi
lines 1-11/11 (END)
```

The date and time formats in this output are the format in which you need to provide dates and times when you are selecting log messages by period, as we shall see shortly.

To see all the metadata that accompanies each log message, use the verbose modifier.

```
sudo journalctl -n 10 -o verbose
```

```
dave@howtogeek:~/work$ sudo journalctl -n 10 -o verbose
```

There [are many possible fields](#), but it is rare for all fields to be present in a message.

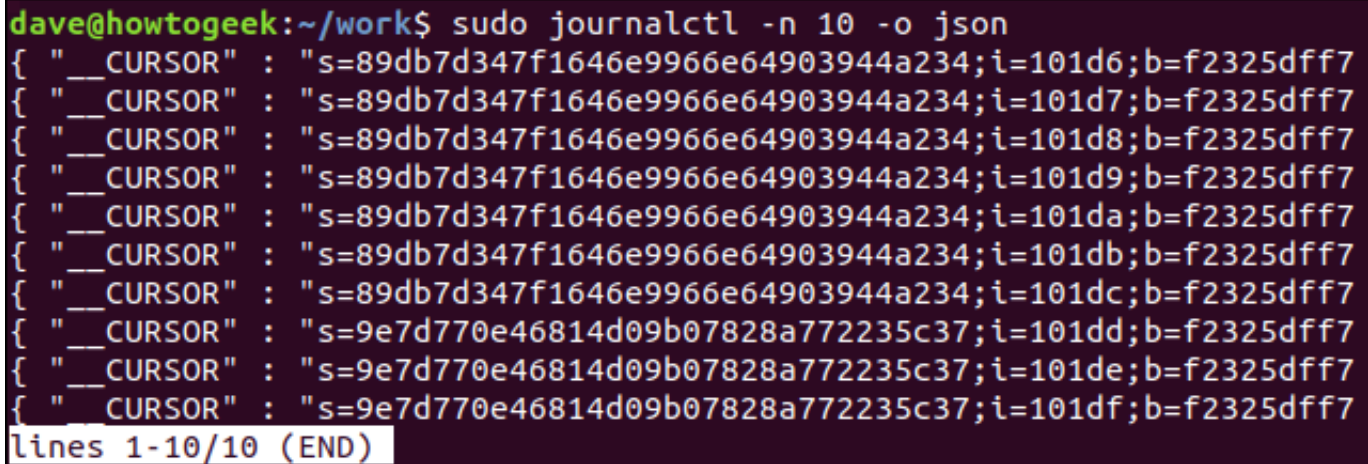
```
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 07
Sun 2020-01-12 07:20:44.154378 EST [s=89db7d347f1646e9966e64903944a234
_UID=1000
_GID=1000
_CAP_EFFECTIVE=0
_SELINUX_CONTEXT=unconfined
_AUDIT_SESSION=2
_AUDIT_LOGINUID=1000
_SYSTEMD_CGROUP=/user.slice/user-1000.slice/session-2.scope
_SYSTEMD_SESSION=2
_SYSTEMD_OWNER_UID=1000
_SYSTEMD_UNIT=session-2.scope
_SYSTEMD_SLICE=user-1000.slice
_SYSTEMD_USER_SLICE=-.slice
_MACHINE_ID=074b4a99814042468da3bc277ada889d
_HOSTNAME=howtogeek
TRANSPORT=stdout
PRIORITY=6
SYSLOG_IDENTIFIER=shutter.desktop
COMM=shutter
lines 1-20
```

One field worth discussing is the `Priority` field. In this example, it has a value of 6. The value represents the importance of the message:

- **0:** Emergency. The system is unusable.
- **1:** Alert. A condition has been flagged that should be corrected immediately.
- **2:** Critical. This covers crashes, coredumps, and significant failures in primary applications.
- **3:** Error. An error has been reported, but it is not considered severe.
- **4:** Warning. Brings a condition to your attention that, if ignored, may become an error.
- **5:** Notice. Used to report events that are unusual, but not errors.
- **6:** Information. Regular operational messages. These do not require action.
- **7:** Debug. Messages put into applications to make it easier for them to debug them.

If you want the output to be presented as properly formed [JavaScript Object Notation](#) (JSON) objects, use the `json` modifier:

```
sudo journalctl -n 10 -o json
```

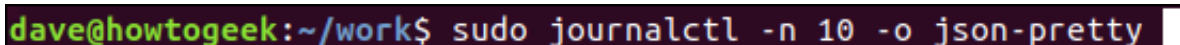


```
dave@howtogeek:~/work$ sudo journalctl -n 10 -o json
{ "__CURSOR" : "s=89db7d347f1646e9966e64903944a234;i=101d6;b=f2325dff7"
{ "__CURSOR" : "s=89db7d347f1646e9966e64903944a234;i=101d7;b=f2325dff7"
{ "__CURSOR" : "s=89db7d347f1646e9966e64903944a234;i=101d8;b=f2325dff7"
{ "__CURSOR" : "s=89db7d347f1646e9966e64903944a234;i=101d9;b=f2325dff7"
{ "__CURSOR" : "s=89db7d347f1646e9966e64903944a234;i=101da;b=f2325dff7"
{ "__CURSOR" : "s=89db7d347f1646e9966e64903944a234;i=101db;b=f2325dff7"
{ "__CURSOR" : "s=89db7d347f1646e9966e64903944a234;i=101dc;b=f2325dff7"
{ "__CURSOR" : "s=9e7d770e46814d09b07828a772235c37;i=101dd;b=f2325dff7"
{ "__CURSOR" : "s=9e7d770e46814d09b07828a772235c37;i=101de;b=f2325dff7"
{ "__CURSOR" : "s=9e7d770e46814d09b07828a772235c37;i=101df;b=f2325dff7"
lines 1-10/10 (END)
```

Each message is properly wrapped as a well-formed JSON object, and displayed one message per line of output.

To have the JSON output [pretty-printed](#), use the `json-pretty` modifier.

```
sudo journalctl -n 10 -o json-pretty
```



```
dave@howtogeek:~/work$ sudo journalctl -n 10 -o json-pretty
```

Each JSON object is split across multiple lines, with each name-value pair on a new line.

```
{
  "__CURSOR" : "s=89db7d347f1646e9966e64903944a234;i=102d2;b=f23
  "__REALTIME_TIMESTAMP" : "1578831756630647",
  "__MONOTONIC_TIMESTAMP" : "10688824642",
  "__BOOT_ID" : "f2325dff7c364ae4bc3bda12277cfb9b",
  "__UID" : "1000",
  "__GID" : "1000",
  "__CAP_EFFECTIVE" : "0",
  "__SELINUX_CONTEXT" : "unconfined\n",
  "__AUDIT_SESSION" : "2",
  "__AUDIT_LOGINUID" : "1000",
  "__SYSTEMD_CGROUP" : "/user.slice/user-1000.slice/session-2.sco
  "__SYSTEMD_SESSION" : "2",
  "__SYSTEMD_OWNER_UID" : "1000",
  "__SYSTEMD_UNIT" : "session-2.scope",
  "__SYSTEMD_SLICE" : "user-1000.slice",
  "__SYSTEMD_USER_SLICE" : "-.slice",
  "__MACHINE_ID" : "074b4a99814042468da3bc277ada889d",
  "__HOSTNAME" : "howtogeek",
  "__TRANSPORT" : "stdout",

```

lines 1-20

To only see the log entry messages, without time stamps or other metadata, use the `cat` modifier:

```
sudo journalctl -n 10 -o cat
```

```
dave@howtogeek:~/work$ sudo journalctl -o cat
```

This display format can make it difficult to identify which process raised the log event, although some messages do contain a clue.



```
GdkPixbuf-LOG **:      Encoding raw at /usr/bin/shutter line 2891.
GdkPixbuf-LOG **:      Dimensions: 16 x 16 at /usr/bin/shutter line 2
GdkPixbuf-LOG **:      Rowstride: 64, Length: 1048 at /usr/bin/shutte
GdkPixbuf-LOG **:      Copy pixels == false at /usr/bin/shutter line
GdkPixbuf-LOG **:      gdk_pixbuf_from_pixdata() called on: at /usr/bin/shu
GdkPixbuf-LOG **:      Encoding raw at /usr/bin/shutter line 2891.
GdkPixbuf-LOG **:      Dimensions: 16 x 16 at /usr/bin/shutter line 2
GdkPixbuf-LOG **:      Rowstride: 64, Length: 1048 at /usr/bin/shutte
GdkPixbuf-LOG **:      Copy pixels == false at /usr/bin/shutter line
INFO: DBus connection to org.freedesktop.compiz failed --> skipping co
org.freedesktop.DBus.Error.ServiceUnknown: The name org.freedesktop.co
GLib-CRITICAL **: Source ID 4653 was not found when attempting to remo
GLib-CRITICAL **: Source ID 4655 was not found when attempting to remo
Window manager warning: Buggy client sent a _NET_ACTIVE_WINDOW message
    dave : TTY=pts/0 ; PWD=/home/dave/work ; USER=root ; COMMAND=/bin/
pam_unix(sudo:session): session opened for user root by (uid=0)
pam_unix(sudo:session): session closed for user root
Third Test Message from HTG
    dave : TTY=pts/0 ; PWD=/home/dave/work ; USER=root ; COMMAND=/bin/
pam_unix(sudo:session): session opened for user root by (uid=0)
lines 61676-61695/61695 (END)
```

## Selecting Log Messages By Time Period

To limit the output from `journalctl` to a time period you're interested in, use the `-s` (since) and `-u` (until) options.

To see the log entries since a particular time and date, use this command:

```
sudo journalctl -S "2020-01-12 07:00:00"
```

```
dave@howtogeek:~/work$ sudo journalctl -S "2020-01-12 07:00:00" |
```

The display contains only messages that arrived after the date and time in the command.

```
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 07:03:14 EDT.
Jan 12 07:03:14 howtogeek kernel: usb 2-1: New USB device found, idVendor=046d:08f4
Jan 12 07:03:14 howtogeek kernel: usb 2-1: New USB device strings: MfrStr=1, Product=1,
Jan 12 07:03:14 howtogeek kernel: usb 2-1: Product: USB Tablet
Jan 12 07:03:14 howtogeek kernel: usb 2-1: Manufacturer: VirtualBox
Jan 12 07:03:14 howtogeek kernel: input: VirtualBox USB Tablet as /dev/input/lm0
Jan 12 07:03:14 howtogeek mtp-probe[10027]: checking bus 2, device 5:
Jan 12 07:03:14 howtogeek mtp-probe[10027]: bus: 2, device: 5 was not found
Jan 12 07:03:14 howtogeek kernel: hid-generic 0003:80EE:0021.0004: input device
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) Configuring input
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) No devices found
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) This input device
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) Configuring input
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) No devices found
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) This input device
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) Configuring input
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (**) VirtualBox USB Tablet
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) Using default libinput
Jan 12 07:03:14 howtogeek upowerd[1312]: unhandled action 'bind' on /sys/devices/pci0000:00/0000:00:02.0/usb2/2-1
Jan 12 07:03:14 howtogeek upowerd[1312]: unhandled action 'bind' on /sys/devices/pci0000:00/0000:00:02.0/usb2/2-1
```

lines 1-20

To define a time period you wish to report on, use both the `-s` (since) and `-u` (until) options together. This command looks at log messages from a 15 minute time period.:

```
sudo journalctl -S "2020-01-12 07:00:00" -U "2020-01-12 07:15:00"
```

```
dave@howtogeek:~/work$ sudo journalctl -S "2020-01-12 07:00:00" -U "2020-01-12 07:15:00"
```

This is a great combination use if you know something odd happened on your system, and roughly when it happened.

```
Jan 12 07:10:47 howtogeek shutter.desktop[9975]: GLib-CRITICAL **: Sou
Jan 12 07:10:47 howtogeek org.gnome.Shell.desktop[1698]: Window manage
Jan 12 07:10:56 howtogeek shutter.desktop[9975]: GLib-CRITICAL **: Sou
Jan 12 07:10:56 howtogeek shutter.desktop[9975]: GLib-GObject-CRITICAL
Jan 12 07:10:58 howtogeek shutter.desktop[9975]: GLib-GObject-CRITICAL
Jan 12 07:11:04 howtogeek nautilus[5974]: gtk_label_set_markup: assert
Jan 12 07:11:04 howtogeek nautilus[5974]: gtk_revealer_set_reveal_chil
Jan 12 07:11:07 howtogeek nautilus[5974]: gtk_revealer_set_reveal_chil
Jan 12 07:11:10 howtogeek eog[10202]: Failed to open file '/home/dave/
Jan 12 07:11:15 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **: gdk
Jan 12 07:11:15 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:11:15 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:11:15 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:11:15 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:11:15 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **: gdk
Jan 12 07:11:15 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:11:15 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:11:15 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:11:15 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 07:11:47 howtogeek sudo[10164]: pam_unix(sudo:session): session
lines 794-813/813 (END)
```

## Using Relative Time Periods

You can use relative addressing when you select your time periods. That means you can say things like “show me all events from one day ago up until now.” This is just what this command means. The “d” stands for “day”, and the “-1” means one day in the past.

```
sudo journalctl -S -1d
```

```
dave@howtogeek:~/work$ sudo journalctl -S -1d
```

The log messages are listed from 00:00:00 yesterday, up until “now.”

```
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 08
Jan 11 16:11:13 howtogeek kernel: 21:11:13.511092 timesync vgsvcTimeSy
Jan 11 16:11:13 howtogeek systemd[1]: Starting Daily apt download acti
Jan 11 16:11:13 howtogeek systemd[1]: Starting Message of the Day...
Jan 11 16:11:13 howtogeek systemd[1]: Started Run anacron jobs.
Jan 11 16:11:13 howtogeek anacron[3740]: Anacron 2.3 started on 2020-0
Jan 11 16:11:13 howtogeek anacron[3740]: Will run job `cron.daily' in
Jan 11 16:11:13 howtogeek anacron[3740]: Jobs will be executed sequent
Jan 11 16:11:13 howtogeek 50-motd-news[3739]: * Overheard at KubeCon:
Jan 11 16:11:13 howtogeek 50-motd-news[3739]:      https://microk8s.io
Jan 11 16:11:13 howtogeek systemd[1]: Started Message of the Day.
Jan 11 16:11:14 howtogeek systemd-resolved[361]: Server returned error
Jan 11 16:11:14 howtogeek systemd-resolved[361]: Server returned error
Jan 11 16:11:23 howtogeek kernel: 21:11:23.511512 timesync vgsvcTimeSy
Jan 11 16:11:25 howtogeek gnome-software[2233]: Only 0 apps for recent
Jan 11 16:11:26 howtogeek gnome-software[2233]: tried overwriting io.s
                                text-shadow: 0 1px 1px
                                color: #000000;
                                outline-offset: 0;
                                outline-color: alpha(#

lines 1-20
```

If you want to investigate something that happened in the recent past, you can specify a relative time period measured in hours. Here we're reviewing log messages from the last hour:

```
sudo journalctl -S -1h
```

```
dave@howtogeek:~/work$ sudo journalctl -S -1h
```

The messages from the last hour are displayed for you. You can also use "m" to set relative time periods measured in minutes, and "w" for weeks.



```
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 08
Jan 12 07:03:14 howtogeek kernel: usb 2-1: New USB device found, idVen
Jan 12 07:03:14 howtogeek kernel: usb 2-1: New USB device strings: Mfr
Jan 12 07:03:14 howtogeek kernel: usb 2-1: Product: USB Tablet
Jan 12 07:03:14 howtogeek kernel: usb 2-1: Manufacturer: VirtualBox
Jan 12 07:03:14 howtogeek kernel: input: VirtualBox USB Tablet as /dev
Jan 12 07:03:14 howtogeek mtp-probe[10027]: checking bus 2, device 5:
Jan 12 07:03:14 howtogeek mtp-probe[10027]: bus: 2, device: 5 was not
Jan 12 07:03:14 howtogeek kernel: hid-generic 0003:80EE:0021.0004: inp
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) conf
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) No i
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) This
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) conf
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) No i
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) This
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) conf
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (**) Virt
Jan 12 07:03:14 howtogeek /usr/lib/gdm3/gdm-x-session[1523]: (II) Usin
Jan 12 07:03:14 howtogeek upowerd[1312]: unhandled action 'bind' on /s
Jan 12 07:03:14 howtogeek upowerd[1312]: unhandled action 'bind' on /s
lines 1-20
```

journalctl understands today, yesterday, and tomorrow. These modifiers provide a handy way to specify common time periods. To see all the events that happened yesterday, use this command:

```
sudo journalctl -S yesterday
```

```
dave@howtogeek:~/work$ sudo journalctl -S yesterday
```

All journal log events that happened yesterday, up to midnight 00:00:00, are retrieved and displayed for you.



```
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 08
Jan 11 16:11:13 howtogeek kernel: 21:11:13.511092 timesync vgsvcTimeSy
Jan 11 16:11:13 howtogeek systemd[1]: Starting Daily apt download acti
Jan 11 16:11:13 howtogeek systemd[1]: Starting Message of the Day...
Jan 11 16:11:13 howtogeek systemd[1]: Started Run anacron jobs.
Jan 11 16:11:13 howtogeek anacron[3740]: Anacron 2.3 started on 2020-0
Jan 11 16:11:13 howtogeek anacron[3740]: Will run job `cron.daily' in
Jan 11 16:11:13 howtogeek anacron[3740]: Jobs will be executed sequent
Jan 11 16:11:13 howtogeek 50-motd-news[3739]: * Overheard at KubeCon:
Jan 11 16:11:13 howtogeek 50-motd-news[3739]:      https://microk8s.io
Jan 11 16:11:13 howtogeek systemd[1]: Started Message of the Day.
Jan 11 16:11:14 howtogeek systemd-resolved[361]: Server returned error
Jan 11 16:11:14 howtogeek systemd-resolved[361]: Server returned error
Jan 11 16:11:23 howtogeek kernel: 21:11:23.511512 timesync vgsvcTimeSy
Jan 11 16:11:25 howtogeek gnome-software[2233]: Only 0 apps for recent
Jan 11 16:11:26 howtogeek gnome-software[2233]: tried overwriting io.s
                                text-shadow: 0 1px 1px
                                color: #000000;
                                outline-offset: 0;
                                outline-color: alpha(#

lines 1-20
```

To see all the log messages received today so far, use this command:

```
sudo journalctl -S today
```

```
dave@howtogeek:~/work$ sudo journalctl -S today
```

Everything from 00:00:00 up until the time the command is issued, are displayed.

```
Jan 12 08:02:31 howtogeek shutter.desktop[9975]: GLib-CRITICAL **: Sou
Jan 12 08:02:31 howtogeek org.gnome.Shell.desktop[1698]: Window manage
Jan 12 08:02:34 howtogeek sudo[10447]: pam_unix(sudo:session): session
Jan 12 08:02:43 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **: gdk
Jan 12 08:02:43 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 08:02:43 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 08:02:43 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 08:02:43 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **: gdk
Jan 12 08:02:43 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 08:02:43 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 08:02:43 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 08:02:43 howtogeek shutter.desktop[9975]: GdkPixbuf-LOG **:
Jan 12 08:02:44 howtogeek shutter.desktop[9975]: INFO: DBus connection
Jan 12 08:02:44 howtogeek shutter.desktop[9975]: org.freedesktop.DBus.
Jan 12 08:02:45 howtogeek shutter.desktop[9975]: GLib-CRITICAL **: Sou
Jan 12 08:02:45 howtogeek shutter.desktop[9975]: GLib-CRITICAL **: Sou
Jan 12 08:02:45 howtogeek org.gnome.Shell.desktop[1698]: Window manage
Jan 12 08:02:46 howtogeek sudo[10453]:      dave : TTY=pts/0 ; PWD=/hom
Jan 12 08:02:46 howtogeek sudo[10453]: pam_unix(sudo:session): session
lines 2324-2343/2343 (END)
```

You're able to mix the different time period modifiers. To see everything from two days ago up until the start of today, use this command:

```
sudo journalctl -S -2d -U today
```

```
dave@howtogeek:~/work$ sudo journalctl -S -2d -U today
```

Everything since the day before yesterday up until today is retrieved and displayed.

```
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 08
Jan 10 09:18:54 howtogeek kernel: 14:18:54.132066 timesync vgsvcTimeSy
Jan 10 09:18:54 howtogeek systemd[1]: Started Run anacron jobs.
Jan 10 09:18:54 howtogeek systemd[1]: Starting Daily apt download acti
Jan 10 09:18:54 howtogeek anacron[12371]: Anacron 2.3 started on 2020-
Jan 10 09:18:54 howtogeek anacron[12371]: Normal exit (0 jobs run)
Jan 10 09:18:54 howtogeek NetworkManager[597]: <info> [1578665934.181
Jan 10 09:18:54 howtogeek NetworkManager[597]: <info> [1578665934.181
Jan 10 09:18:54 howtogeek kernel: e1000: enp0s3 NIC Link is Up 1000 Mb
Jan 10 09:18:54 howtogeek NetworkManager[597]: <info> [1578665934.190
Jan 10 09:18:54 howtogeek NetworkManager[597]: <info> [1578665934.190
Jan 10 09:18:54 howtogeek NetworkManager[597]: <info> [1578665934.191
Jan 10 09:18:54 howtogeek NetworkManager[597]: <info> [1578665934.193
Jan 10 09:18:54 howtogeek dhclient[12394]: DHCPDISCOVER on enp0s3 to 2
Jan 10 09:18:54 howtogeek dhclient[12394]: DHCPREQUEST of 192.168.4.26
Jan 10 09:18:54 howtogeek dhclient[12394]: DHCP OFFER of 192.168.4.26 f
Jan 10 09:18:54 howtogeek dhclient[12394]: DHCPACK of 192.168.4.26 fro
Jan 10 09:18:54 howtogeek NetworkManager[597]: <info> [1578665934.222
Jan 10 09:18:54 howtogeek NetworkManager[597]: <info> [1578665934.222
Jan 10 09:18:54 howtogeek NetworkManager[597]: <info> [1578665934.222
lines 1-20
```

## Selecting Log Messages By Data Fields

You can search for log messages [that match a wide range of journal fields](#). These searches try to find matches in the metadata attached to each message. It is recommended that you [refer to the list of fields](#) and choose the ones that will be most useful to you.

Bear in mind, whether an application completes every field or not is entirely up to the authors of the application. You can't guarantee every field will be populated.

All of the journal field modifiers are used in the same way. We'll use a few in our examples below. To look for log messages from a specific application, use the `_COMM` (command) modifier. If you also use the `-f` (follow) option, `journalctl` will track new messages from this application as they arrive.

```
sudo journalctl -f _COMM=geek-app
```

```
dave@howtogeek:~/work$ sudo journalctl -f _COMM=geek-app
-- Logs begin at Mon 2019-09-30 14:55:07 EDT. --
Jan 11 17:28:41 howtogeek geek-app[8620]: Hello World!
Jan 11 17:54:14 howtogeek geek-app[8885]: Hello World!
Jan 11 17:56:45 howtogeek geek-app[8896]: Hello World!
Jan 12 07:23:21 howtogeek geek-app[10297]: Second Test Message from HT
G
Jan 12 07:25:45 howtogeek geek-app[10314]: Third Test Message from HTG
Jan 12 08:10:18 howtogeek geek-app[10509]: Fourth Test Message from HT
G
█
```

You can search for log entries using the [process ID](#) of the process that generated the log message. Use the `ps` command to find the [process id of the daemon or application you're going to search for](#).

```
sudo journalctl _PID=751
```

```
dave@howtogeek:~/work$ sudo journalctl _PID=751 █
```

On the machine used to research this article, the [SSH](#) daemon is process 751.

```
dave@howtogeek:~/work$ sudo journalctl _PID=751
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 08
Jan 10 16:33:17 howtogeek sshd[751]: Server listening on 0.0.0.0 port
Jan 10 16:33:17 howtogeek sshd[751]: Server listening on :: port 22.
Jan 10 16:33:19 howtogeek sshd[751]: Received SIGHUP; restarting.
Jan 10 16:33:19 howtogeek sshd[751]: Server listening on 0.0.0.0 port
Jan 10 16:33:19 howtogeek sshd[751]: Server listening on :: port 22.
Jan 10 16:33:19 howtogeek sshd[751]: Received SIGHUP; restarting.
Jan 10 16:33:19 howtogeek sshd[751]: Server listening on 0.0.0.0 port
Jan 10 16:33:19 howtogeek sshd[751]: Server listening on :: port 22.
lines 1-9/9 (END)
```

You can also search by [user id](#). This is the user ID for the person who launched the application or command, or who owns the process.

```
sudo journalctl _UID=1000
```

```
dave@howtogeek:~/work$ sudo journalctl _UID=1000
```

All messages associated with any other user ID's are filtered out. Only messages related to user 1000 are shown:



```
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 08
Sep 30 14:55:52 howtogeek systemd[2673]: Listening on GnuPG cryptograp
Sep 30 14:55:52 howtogeek systemd[2673]: Started Pending report trigge
Sep 30 14:55:52 howtogeek systemd[2673]: Listening on GnuPG cryptograp
Sep 30 14:55:52 howtogeek systemd[2673]: Listening on GnuPG network ce
Sep 30 14:55:52 howtogeek systemd[2673]: Reached target Paths.
Sep 30 14:55:52 howtogeek systemd[2673]: Reached target Timers.
Sep 30 14:55:52 howtogeek systemd[2673]: Listening on GnuPG cryptograp
Sep 30 14:55:52 howtogeek systemd[2673]: Listening on GnuPG cryptograp
Sep 30 14:55:52 howtogeek systemd[2673]: Starting D-Bus User Message B
Sep 30 14:55:52 howtogeek systemd[2673]: Listening on D-Bus User Messa
Sep 30 14:55:52 howtogeek systemd[2673]: Reached target Sockets.
Sep 30 14:55:52 howtogeek systemd[2673]: Reached target Basic System.
Sep 30 14:55:52 howtogeek systemd[2673]: Reached target Default.
Sep 30 14:55:52 howtogeek systemd[2673]: Startup finished in 19ms.
Sep 30 14:55:53 howtogeek /usr/lib/gdm3/gdm-x-session[2691]: (--) Log
Sep 30 14:55:53 howtogeek /usr/lib/gdm3/gdm-x-session[2691]: X.Org X S
Sep 30 14:55:53 howtogeek /usr/lib/gdm3/gdm-x-session[2691]: X Protoco
Sep 30 14:55:53 howtogeek /usr/lib/gdm3/gdm-x-session[2691]: Build Ope
Sep 30 14:55:53 howtogeek /usr/lib/gdm3/gdm-x-session[2691]: Current 0
lines 1-20
```

Another way to search for log messages related to a specific application is to provide the path to the executable.

```
sudo journalctl /usr/bin/anacron
```

```
dave@howtogeek:~/work$ sudo journalctl /usr/sbin/anacron
```

All of the `anacron` [scheduler log messages](#) are retrieved and displayed.

```
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 08
Sep 30 14:55:07 howtogeek anacron[673]: Anacron 2.3 started on 2019-09
Sep 30 14:55:07 howtogeek anacron[673]: Will run job `cron.daily' in 5
Sep 30 14:55:07 howtogeek anacron[673]: Will run job `cron.weekly' in
Sep 30 14:55:07 howtogeek anacron[673]: Will run job `cron.monthly' in
Sep 30 14:55:07 howtogeek anacron[673]: Jobs will be executed sequenti
-- Reboot --
Sep 30 15:10:45 howtogeek anacron[509]: Anacron 2.3 started on 2019-09
Sep 30 15:10:45 howtogeek anacron[509]: Normal exit (0 jobs run)
-- Reboot --
Sep 30 15:59:58 howtogeek anacron[673]: Job `cron.daily' started
Sep 30 16:00:01 howtogeek anacron[673]: Job `cron.daily' terminated
Sep 30 16:00:01 howtogeek anacron[673]: Job `cron.weekly' started
Sep 30 16:00:01 howtogeek anacron[673]: Job `cron.weekly' terminated
Sep 30 16:00:01 howtogeek anacron[673]: Job `cron.monthly' started
Sep 30 16:00:01 howtogeek anacron[673]: Job `cron.monthly' terminated
Sep 30 16:00:01 howtogeek anacron[673]: Normal exit (3 jobs run)
-- Reboot --
Oct 01 17:22:13 howtogeek anacron[511]: Anacron 2.3 started on 2019-10
Oct 01 17:22:13 howtogeek anacron[511]: Will run job `cron.daily' in 5
lines 1-20
```

To make searching easier, we can ask `journalctl` to list all the values it holds, for any of the journal fields.

To see the user ID's that `journalctl` has recorded log messages for, use the `-F` (fields) option, and pass the `_UID` field identifier.

```
journalctl -F _UID
```

```
dave@howtogeek:~/work$ journalctl -F _UID
112
117
109
121
102
116
103
101
0
1000
62583
1002
1
123
dave@howtogeek:~/work$
```

Let's do that again and look at the [group IDs](#) (GID's):

```
journalctl -F _GID
```

```
dave@howtogeek:~/work$ journalctl -F _GID
1000
117
123
114
125
106
122
107
103
0
62583
1002
1
127
dave@howtogeek:~/work$
```

You can do this with any of the [journal field identifiers](#).

## Listing Kernel Messages

There's a built-in way to isolate kernel messages quickly. You don't need to search and isolate them yourself. The `-k` (kernel) option removes all other messages and gives you an instant view of the kernel log entries.

```
sudo journalctl -k
```

```
dave@howtogeek:~/work$ sudo journalctl -k
```

The highlighting reflects the importance of the message, according to the values in the `Priority` field.

```
Jan 11 17:58:05 howtogeek kernel: hid-generic 0003:80EE:0021.0003: inp
Jan 11 17:58:06 howtogeek kernel: e1000: enp0s3 NIC Link is Down
Jan 11 17:58:06 howtogeek kernel: e1000 0000:00:03.0 enp0s3: Reset ada
Jan 11 17:58:08 howtogeek kernel: e1000: enp0s3 NIC Link is Up 1000 Mb
Jan 12 06:34:19 howtogeek kernel: 11:34:19.067079 timesync vgsvcTimeSy
Jan 12 06:34:29 howtogeek kernel: 11:34:29.067211 timesync vgsvcTimeSy
Jan 12 06:52:38 howtogeek kernel: usb 2-1: USB disconnect, device numb
Jan 12 06:52:38 howtogeek kernel: 11:52:38.515598 control Session 0 i
Jan 12 06:52:38 howtogeek kernel: 11:52:38.522032 control Stopping al
Jan 12 06:52:38 howtogeek kernel: 11:52:38.522060 control Closing all
Jan 12 06:52:38 howtogeek kernel: e1000: enp0s3 NIC Link is Down
Jan 12 06:52:38 howtogeek kernel: e1000 0000:00:03.0 enp0s3: Reset ada
Jan 12 06:52:39 howtogeek kernel: usb 2-1: new full-speed USB device n
Jan 12 07:03:14 howtogeek kernel: usb 2-1: New USB device found, idVen
Jan 12 07:03:14 howtogeek kernel: usb 2-1: New USB device strings: Mfr
Jan 12 07:03:14 howtogeek kernel: usb 2-1: Product: USB Tablet
Jan 12 07:03:14 howtogeek kernel: usb 2-1: Manufacturer: VirtualBox
Jan 12 07:03:14 howtogeek kernel: input: VirtualBox USB Tablet as /dev
Jan 12 07:03:14 howtogeek kernel: hid-generic 0003:80EE:0021.0004: inp
Jan 12 07:03:15 howtogeek kernel: e1000: enp0s3 NIC Link is Up 1000 Mb
lines 568-587/587 (END)
```

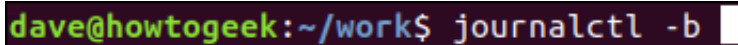


# Reviewing Boot Messages

If you've got an issue related to booting that you wish to investigate, `journalctl` has you covered. Perhaps you've added new hardware, and it isn't responding, or a previously working hardware component no longer works after your last system upgrade.

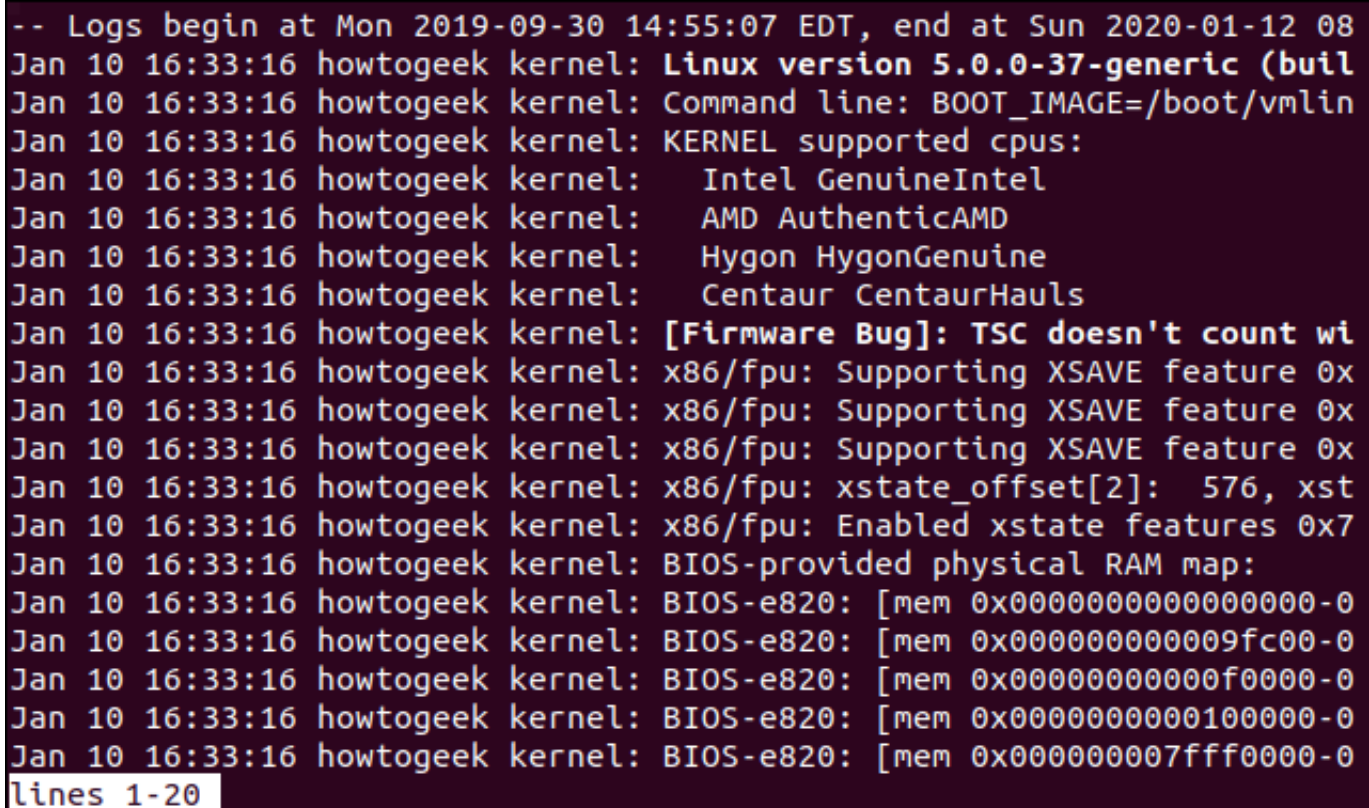
To see the log entries related to your last boot, use the `-b` (boot) option:

```
journalctl -b
```



```
dave@howtogeek:~/work$ journalctl -b
```

The log entries for the last boot are shown for you.

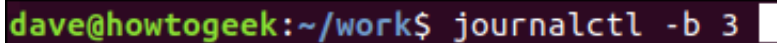


```
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 08
Jan 10 16:33:16 howtogeek kernel: Linux version 5.0.0-37-generic (buil
Jan 10 16:33:16 howtogeek kernel: Command line: BOOT_IMAGE=/boot/vmlin
Jan 10 16:33:16 howtogeek kernel: KERNEL supported cpus:
Jan 10 16:33:16 howtogeek kernel: Intel GenuineIntel
Jan 10 16:33:16 howtogeek kernel: AMD AuthenticAMD
Jan 10 16:33:16 howtogeek kernel: Hygon HygonGenuine
Jan 10 16:33:16 howtogeek kernel: Centaur CentaurHauls
Jan 10 16:33:16 howtogeek kernel: [Firmware Bug]: TSC doesn't count wi
Jan 10 16:33:16 howtogeek kernel: x86/fpu: Supporting XSAVE feature 0x
Jan 10 16:33:16 howtogeek kernel: x86/fpu: Supporting XSAVE feature 0x
Jan 10 16:33:16 howtogeek kernel: x86/fpu: Supporting XSAVE feature 0x
Jan 10 16:33:16 howtogeek kernel: x86/fpu: xstate_offset[2]: 576, xst
Jan 10 16:33:16 howtogeek kernel: x86/fpu: Enabled xstate features 0x7
Jan 10 16:33:16 howtogeek kernel: BIOS-provided physical RAM map:
Jan 10 16:33:16 howtogeek kernel: BIOS-e820: [mem 0x0000000000000000-0
Jan 10 16:33:16 howtogeek kernel: BIOS-e820: [mem 0x0000000000009fc0-0
Jan 10 16:33:16 howtogeek kernel: BIOS-e820: [mem 0x000000000000f000-0
Jan 10 16:33:16 howtogeek kernel: BIOS-e820: [mem 0x0000000000010000-0
Jan 10 16:33:16 howtogeek kernel: BIOS-e820: [mem 0x0000000007fff000-0
lines 1-20
```

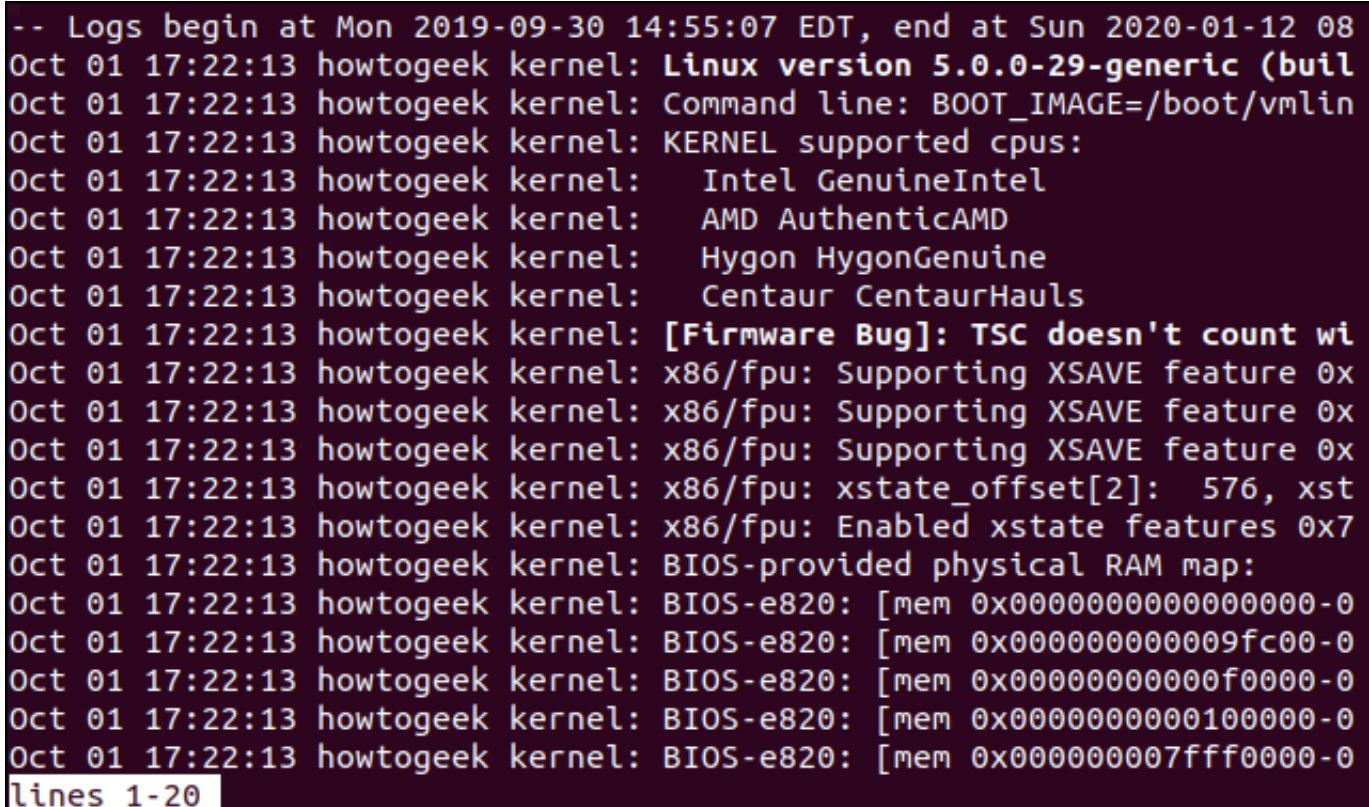


When we say “last boot,” we mean the boot process that brought your computer to life for your current logged-in session. To see previous boots, you can use a number to tell `journalctl` which boot you’re interested in. To see the third previous boot, use this command:

```
journalctl -b 3
```

A terminal window with a dark purple background. The prompt is 'dave@howtogeek:~/work\$' in green. The command 'journalctl -b 3' is entered in white, followed by a white cursor bar.

Generally, if you’ve had a problem and had to reboot your machine, it is a previous boot sequence you’re interested in. So this is a common command form.

A terminal window showing the output of 'journalctl -b 3'. The output is in white text on a dark purple background. It starts with '-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 08'. The log entries are from 'Oct 01 17:22:13 howtogeek kernel:'. The entries include 'Linux version 5.0.0-29-generic (buil', 'Command line: BOOT\_IMAGE=/boot/vmlin', 'KERNEL supported cpus:', 'Intel GenuineIntel', 'AMD AuthenticAMD', 'Hygon HygonGenuine', 'Centaur CentaurHauls', '[Firmware Bug]: TSC doesn't count wi', 'x86/fpu: Supporting XSAVE feature 0x', 'x86/fpu: Supporting XSAVE feature 0x', 'x86/fpu: Supporting XSAVE feature 0x', 'x86/fpu: xstate\_offset[2]: 576, xst', 'x86/fpu: Enabled xstate features 0x7', 'BIOS-provided physical RAM map:', 'BIOS-e820: [mem 0x0000000000000000-0', 'BIOS-e820: [mem 0x00000000000009fc00-0', 'BIOS-e820: [mem 0x000000000000f0000-0', 'BIOS-e820: [mem 0x0000000000100000-0', 'BIOS-e820: [mem 0x000000007fff0000-0'. At the bottom, 'lines 1-20' is highlighted in a white box.

It is easy to get mixed up with the sequence of boots. To help, we can ask `journalctl` to list the boots that it has recorded in its journal, using the --

`list-boots` option.

```
journalctl --list-boots
```

```
dave@howtogeek:~/work$ journalctl --list-boots
-9 189e080ca99147bfada26d7ef3cf57c4 Mon 2019-09-30 14:55:07 EDT-Mon 20
-8 77331fee34e0440398e6d7796e4e3222 Mon 2019-09-30 16:10:37 EDT-Mon 20
-7 1c60a0ce4ab344099790683eba57b6e6 Tue 2019-10-01 17:22:13 EDT-Wed 20
-6 76e0891feb2041b0ac6c9bd9be42aac3 Wed 2019-10-02 05:39:16 EDT-Wed 20
-5 1f00248226ed4ab9a1abac86e0d540d7 Fri 2019-10-04 09:58:27 EDT-Wed 20
-4 81bfb2c2ae134af292acacc317f26c93 Wed 2019-12-04 05:58:04 EST-Wed 20
-3 71ed38b41e1b47c5bc928f7de766e6e6 Wed 2020-01-01 08:31:31 EST-Mon 20
-2 3fc3da55e5b648c5a704a8d30aa818ae Mon 2020-01-06 15:44:33 EST-Mon 20
-1 391f38fd214b4396b6016edb17d05d0c Mon 2020-01-06 16:21:34 EST-Fri 20
 0 f2325dff7c364ae4bc3bda12277cfb9b Fri 2020-01-10 16:33:16 EST-Sun 20
lines 1-10/10 (END)
```

You can identify the boot you wish to see messages for from the date and time stamp, and then use the number in the left-hand column to obtain the log messages for that boot sequence. You can also pick the 32-bit boot identifier, and pass that to `journalctl`.

```
sudo journalctl -b 1f00248226ed4ab9a1abac86e0d540d7
```

```
dave@howtogeek:~/work$ sudo journalctl -b 1f00248226ed4ab9a1abac86e0d5
40d7
```

The log messages from the boot sequence we requested are retrieved and displayed.

```
-- Logs begin at Mon 2019-09-30 14:55:07 EDT, end at Sun 2020-01-12 08
Oct 04 09:58:27 howtogeek kernel: Linux version 5.0.0-29-generic (buil
Oct 04 09:58:27 howtogeek kernel: Command line: BOOT_IMAGE=/boot/vmlin
Oct 04 09:58:27 howtogeek kernel: KERNEL supported cpus:
Oct 04 09:58:27 howtogeek kernel: Intel GenuineIntel
Oct 04 09:58:27 howtogeek kernel: AMD AuthenticAMD
Oct 04 09:58:27 howtogeek kernel: Hygon HygonGenuine
Oct 04 09:58:27 howtogeek kernel: Centaur CentaurHauls
Oct 04 09:58:27 howtogeek kernel: [Firmware Bug]: TSC doesn't count wi
Oct 04 09:58:27 howtogeek kernel: x86/fpu: Supporting XSAVE feature 0x
Oct 04 09:58:27 howtogeek kernel: x86/fpu: Supporting XSAVE feature 0x
Oct 04 09:58:27 howtogeek kernel: x86/fpu: Supporting XSAVE feature 0x
Oct 04 09:58:27 howtogeek kernel: x86/fpu: xstate_offset[2]: 576, xst
Oct 04 09:58:27 howtogeek kernel: x86/fpu: Enabled xstate features 0x7
Oct 04 09:58:27 howtogeek kernel: BIOS-provided physical RAM map:
Oct 04 09:58:27 howtogeek kernel: BIOS-e820: [mem 0x0000000000000000-0
Oct 04 09:58:27 howtogeek kernel: BIOS-e820: [mem 0x0000000000009fc0-0
Oct 04 09:58:27 howtogeek kernel: BIOS-e820: [mem 0x000000000000f000-0
Oct 04 09:58:27 howtogeek kernel: BIOS-e820: [mem 0x0000000000010000-0
Oct 04 09:58:27 howtogeek kernel: BIOS-e820: [mem 0x0000000007fff0000-0
lines 1-20
```

## Managing Journal Hard Drive Space

Of course, the journal and all of its log messages are stored on your hard drive. That means they'll be taking up hard drive space. To see how much space has been taken by the journal, use the `--disk-usage` option.

```
journalctl --disk-usage
```

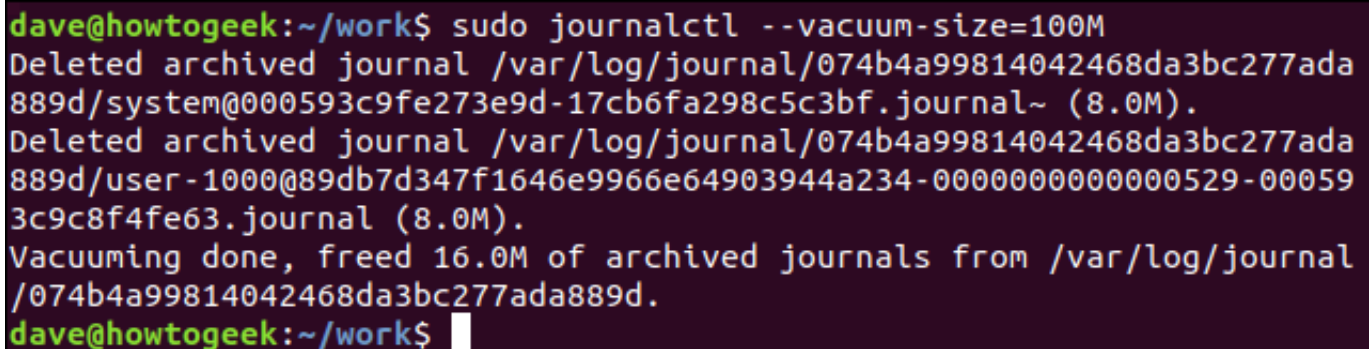
```
dave@howtogeek:~/work$ journalctl --disk-usage
Archived and active journals take up 152.0M in the file system.
dave@howtogeek:~/work$
```

With today's hard drives, 152 MB isn't much space at all, but for demonstration purposes, we'll still trim it back. There's two way we can do this. The first is to set a size limit that you want the journal reduced back to.

It'll grow again, of course, but we can prune it now ready for that new growth.

We'll use the wonderfully titled `--vacuum-size` option, and pass in the size we'd like the journal reduced to. We'll ask for 100 MB. The way to think of this is we're asking `journalctl` to "throw away whatever you can, but don't go lower than 100 MB."

```
journalctl --vacuum-size=100M
```

A terminal window with a dark purple background. The prompt is 'dave@howtogeek:~/work\$'. The command 'sudo journalctl --vacuum-size=100M' has been executed. The output shows two archived journals being deleted: one for the system (8.0M) and one for a user (8.0M). The final output states 'Vacuuming done, freed 16.0M of archived journals from /var/log/journal/074b4a99814042468da3bc277ada889d.' followed by the prompt 'dave@howtogeek:~/work\$' with a cursor.

```
dave@howtogeek:~/work$ sudo journalctl --vacuum-size=100M
Deleted archived journal /var/log/journal/074b4a99814042468da3bc277ada889d/system@000593c9fe273e9d-17cb6fa298c5c3bf.journal~ (8.0M).
Deleted archived journal /var/log/journal/074b4a99814042468da3bc277ada889d/user-1000@89db7d347f1646e9966e64903944a234-00000000000000529-000593c9c8f4fe63.journal (8.0M).
Vacuuming done, freed 16.0M of archived journals from /var/log/journal/074b4a99814042468da3bc277ada889d.
dave@howtogeek:~/work$
```

The other way to trim back the journal size is to use the `--vacuum-time` option. This option tells `journalctl` to discard messages that are older than the period you provide on the command line. You can use `days`, `weeks`, `months`, and `years` in the time period.

Let's weed out all messages older than one week:

```
journalctl --vacuum-time=1weeks
```



```
dave@howtogeek:~/work$ sudo journalctl --vacuum-time=1weeks
Deleted archived journal /var/log/journal/074b4a99814042468da3bc277ada889d/system@cc3c7788073249eaba5c78292212240d-0000000000000001-000593c9fe268da2.journal (16.0M).
Deleted archived journal /var/log/journal/074b4a99814042468da3bc277ada889d/system@000594223083cfa9-2b8a0eb68564e312.journal~ (8.0M).
Deleted archived journal /var/log/journal/074b4a99814042468da3bc277ada889d/system@000598deafde7b89-50ec164cb22e0a80.journal~ (8.0M).
Deleted archived journal /var/log/journal/074b4a99814042468da3bc277ada889d/user-1000@89db7d347f1646e9966e64903944a234-00000000000002acb-0005985a6408506c.journal (24.0M).
Deleted archived journal /var/log/journal/074b4a99814042468da3bc277ada889d/system@3b2fe004e4b6437ea5d50718d2ec2108-0000000000000001-000598deafddbfa8.journal (16.0M).
Deleted archived journal /var/log/journal/074b4a99814042468da3bc277ada889d/user-1002@37d1dd731e00452db4aa3e6b9667ff53-000000000000082f9-000599fb2fda2dac.journal (8.0M).
Deleted archived journal /var/log/journal/074b4a99814042468da3bc277ada889d/system@00059b7eba150cf8-086fa4961c091d23.journal~ (8.0M).
Vacuuming done, freed 88.0M of archived journals from /var/log/journal/074b4a99814042468da3bc277ada889d.
```

## Data vs. Information

Data isn't useful unless you can get at it and make use of it. Then it becomes useful information. The `journalctl` command is a flexible and sophisticated tool that allows you to get to the information of interest in a variety of ways.

You can use just about any snippet of information you have to home in on the log messages you need.

### READ NEXT

- › [How to Customize Automatic Updates on iPhone and iPad](#)
- › [Can You Move Windows 10's Notification Pop-ups?](#)
- › [CPUs Decoded: Understanding Intel's Microarchitecture Names](#)
- › [How to Scan a File or Folder for Malware with Microsoft Defender on Windows 10](#)



› [How to Use the whois Command on Linux](#)