

scancel(1)

Slurm Commands

scancel(1)

**NAME**

scancel – Used to signal jobs or job steps that are under the control of Slurm.

**SYNOPSIS**

**scancel** [*OPTIONS...*] [*job\_id[\_array\_id][.step\_id]*] [*job\_id[\_array\_id][.step\_id]*...]

**DESCRIPTION**

**scancel** is used to signal or cancel jobs, job arrays or job steps. An arbitrary number of jobs or job steps may be signaled using job specification filters or a space separated list of specific job and/or job step IDs. If the job ID of a job array is specified with an array ID value then only that job array element will be cancelled. If the job ID of a job array is specified without an array ID value then all job array elements will be cancelled. While a heterogeneous job is in pending state, only the entire job can be cancelled rather than it's individual components. A request to cancel an individual component of a heterogeneous job not in pending state will return an error. After the job has begun execution, the individual component can be cancelled. A job or job step can only be signaled by the owner of that job or user root. If an attempt is made by an unauthorized user to signal a job or job step, an error message will be printed and the job will not be signaled.

**OPTIONS**

**-A, --account=*account***

Restrict the scancel operation to jobs under this charge account.

**-b, --batch**

By default, signals other than SIGKILL are not sent to the batch step (the shell script). With this option **scancel** signals only the batch step, but not any other steps. This is useful when the shell script has to trap the signal and take some application defined action. Note that most shells cannot handle signals while a command is running (child process of the batch step), the shell use to wait until the command ends to then handle the signal. Children of the batch step are not signaled with this option, use **-f, --full** instead. NOTE: If used with **-f, --full**, this option ignored. NOTE: This option is not applicable if *step\_id* is specified. NOTE: The shell itself may exit upon receipt of many signals. You may avoid this by explicitly trap signals within the shell script (e.g. "trap <arg> <signals>"). See the shell documentation for details.

**--ctld** Send the job signal request to the slurmd daemon rather than directly to the slurmd daemons. This increases overhead, but offers better fault tolerance. This is the default behavior on architectures using front end nodes (e.g. Cray ALPS computers) or when the **--clusters** option is used.

**-f, --full**

By default, signals other than SIGKILL are not sent to the batch step (the shell script). With this option **scancel** signals also the batch script and its children processes. Most shells cannot handle signals while a command is running (child process of the batch step), the shell use to wait until the command ends to then handle the signal. Unlike **-b, --batch**, children of the batch step are also signaled with this option. NOTE: srun steps are also children of the batch step, so steps are also signaled with this option.

**--help** Print a help message describing all **scancel** options.

**-H, --hurry**

Do not stage out any burst buffer data.

**-i, --interactive**

Interactive mode. Confirm each *job\_id.step\_id* before performing the cancel operation.



- M, --clusters=<string>**  
Clusters to issue commands to. Note that the SlurmDBD must be up for this option to work properly.
- n, --jobname=job\_name, --name=job\_name**  
Restrict the scancel operation to jobs with this job name.
- p, --partition=partition\_name**  
Restrict the scancel operation to jobs in this partition.
- q, --qos=qos**  
Restrict the scancel operation to jobs with this quality of service.
- Q, --quiet**  
Do not report an error if the specified job is already completed. This option is incompatible with the **--verbose** option.
- R, --reservation=reservation\_name**  
Restrict the scancel operation to jobs with this reservation name.
- sibling=cluster\_name**  
Remove an active sibling job from a federated job.
- s, --signal=signal\_name**  
The name or number of the signal to send. If this option is not used the specified job or step will be terminated. **Note.** If this option is used the signal is sent directly to the slurmd where the job is running bypassing the slurmctld thus the job state will not change even if the signal is delivered to it. Use the *scontrol* command if you want the job state change be known to slurmctld.
- t, --state=job\_state\_name**  
Restrict the scancel operation to jobs in this state. *job\_state\_name* may have a value of either "PENDING", "RUNNING" or "SUSPENDED".
- u, --user=user\_name**  
Restrict the scancel operation to jobs owned by this user.
- usage**  
Print a brief help message listing the **scancel** options.
- v, --verbose**  
Print additional logging. Multiple v's increase logging detail. This option is incompatible with the **--quiet** option.
- V, --version**  
Print the version number of the scancel command.
- w, --odelist=host1,host2,...**  
Cancel any jobs using any of the given hosts. The list may be specified as a comma-separated list of hosts, a range of hosts (host[1-5,7,...] for example), or a filename. The host list will be assumed to be a filename only if it contains a "/" character.



**--wckey=wckey**

Restrict the scancel operation to jobs using this workload characterization key.

## ARGUMENTS

*job\_id* The Slurm job ID to be signaled.

*step\_id* The step ID of the job step to be signaled. If not specified, the operation is performed at the level of a job.

If neither **--batch** nor **--signal** are used, the entire job will be terminated.

When **--batch** is used, the batch shell processes will be signaled. The child processes of the shell will not be signaled by Slurm, but the shell may forward the signal.

When **--batch** is not used but **--signal** is used, then all job steps will be signaled, but the batch script itself will not be signaled.

## ENVIRONMENT VARIABLES

Some **scancel** options may be set via environment variables. These environment variables, along with their corresponding options, are listed below. (Note: commandline options will always override these settings)

### SCANCEL\_ACCOUNT

**-A, --account=account**

**SCANCEL\_BATCH** **-b, --batch**

**SCANCEL\_CTLD** **--ctld**

**SCANCEL\_FULL** **-f, --full**

**SCANCEL\_HURRY** **-H, --hurry**

### SCANCEL\_INTERACTIVE

**-i, --interactive**

**SCANCEL\_NAME** **-n, --name=job\_name**

### SCANCEL\_PARTITION

**-p, --partition=partition\_name**

**SCANCEL\_QOS** **-q, --qos=qos**

**SCANCEL\_STATE** **-t, --state=job\_state\_name**

**SCANCEL\_USER** **-u, --user=user\_name**

### SCANCEL\_VERBOSE

**-v, --verbose**

**SCANCEL\_WCKEY** **--wckey=wckey**

**SLURM\_CONF** The location of the Slurm configuration file.

## NOTES

If multiple filters are supplied (e.g. **--partition** and **--name**) only the jobs satisfying all of the filtering options will be signaled.

Cancelling a job step will not result in the job being terminated. The job must be cancelled to release a resource allocation.

To cancel a job, invoke **scancel** without **--signal** option. This will send first a SIGCONT to all steps to eventually wake them up followed by a SIGTERM, then wait the KillWait duration defined in the slurm.conf file and finally if they have not terminated send a SIGKILL. This gives time for the running job/step(s) to clean up.



If a signal value of "KILL" is sent to an entire job, this will cancel the active job steps but not cancel the job itself.

On Cray systems, all signals **except** SIGCHLD, SIGCONT, SIGSTOP, SIGTSTP, SIGTTIN, SIGTTOU, SIGURG, or SIGWINCH cause the ALPS reservation to be released. The job however will not be terminated except in the case of SIGKILL and may then be used for post processing.

## AUTHORIZATION

When using the Slurm db, users who have AdminLevel's defined (Operator or Admin) and users who are account coordinators are given the authority to invoke scancel on other user's jobs.

## EXAMPLES

Send SIGTERM to steps 1 and 3 of job 1234:

```
scancel --signal=TERM 1234.1 1234.3
```

Cancel job 1234 along with all of its steps:

```
scancel 1234
```

Send SIGKILL to all steps of job 1235, but do not cancel the job itself:

```
scancel --signal=KILL 1235
```

Send SIGUSR1 to the batch shell processes of job 1236:

```
scancel --signal=USR1 --batch 1236
```

Cancel job all pending jobs belonging to user "bob" in partition "debug":

```
scancel --state=PENDING --user=bob --partition=debug
```

Cancel only array ID 4 of job array 1237

```
scancel 1237_4
```

## COPYING

Copyright (C) 2002-2007 The Regents of the University of California. Produced at Lawrence Livermore National Laboratory (cf, DISCLAIMER).

Copyright (C) 2008-2011 Lawrence Livermore National Security.

Copyright (C) 2010-2015 SchedMD LLC.

This file is part of Slurm, a resource management program. For details, see <<https://slurm.schedmd.com/>>.

Slurm is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

Slurm is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

## SEE ALSO

**slurm\_kill\_job** (3), **slurm\_kill\_job\_step** (3)

