

# Torque

## My job exited with a code and I do not know what that code refers to.

**Issue:** My job exited with a code, and I do not know what that code refers to.

**Affected Versions:** ALL

**Symptom:** E.g.

```
qstat -f "exit_status = 166"
```

```
checkjob "Completion Code: 166"
```

### Solution:

These values are Linux exit codes, generally returned by the application. For example, if I wrote a bash script to exit 126, then the checkjob output would display 126. Let's say that the job failed early on in bash itself. In that case, the exit code bash returned would get passed up. Here is a complete list. Note these are all external to Moab and TORQUE.

In this case you received

[http://en.wikipedia.org/wiki/Unix\\_signal](http://en.wikipedia.org/wiki/Unix_signal)

### SIGWINCH

The SIGWINCH signal is sent to a process when its controlling terminal changes its size (a window change).

```
root#> kill -l
```

- 1) SIGHUP 2) SIGINT 3) SIGQUIT 4) SIGILL 5) SIGTRAP
- 6) SIGABRT 7) SIGBUS 8) SIGFPE 9) SIGKILL 10) SIGUSR1
- 11) SIGSEGV 12) SIGUSR2 13) SIGPIPE 14) SIGALRM 15) SIGTERM

# Torque

16) SIGSTKFLT 17) SIGCHLD 18) SIGCONT 19) SIGSTOP 20) SIGTSTP  
21) SIGTTIN 22) SIGTTOU 23) SIGURG 24) SIGXCPU 25) SIGXFSZ  
26) SIGVTALRM 27) SIGPROF 28) SIGWINCH 29) SIGIO 30) SIGPWR  
31) SIGSYS 34) SIGRTMIN 35) SIGRTMIN+1 36) SIGRTMIN+2 37) SIGRTMIN+3  
38) SIGRTMIN+4 39) SIGRTMIN+5 40) SIGRTMIN+6 41) SIGRTMIN+7 42)  
SIGRTMIN+8  
43) SIGRTMIN+9 44) SIGRTMIN+10 45) SIGRTMIN+11 46) SIGRTMIN+12 47)  
SIGRTMIN+13  
48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14 51) SIGRTMAX-13 52)  
SIGRTMAX-12  
53) SIGRTMAX-11 54) SIGRTMAX-10 55) SIGRTMAX-9 56) SIGRTMAX-8 57)  
SIGRTMAX-7  
58) SIGRTMAX-6 59) SIGRTMAX-5 60) SIGRTMAX-4 61) SIGRTMAX-3 62) SIGRTMAX-2  
63) SIGRTMAX-1 64) SIGRTMAX

man 7 signal

(or, on Solaris, "man \-s 3HEAD signal"). This will give you the man page for SIGNAL(7). Scroll down a bit and you will get a list of the kill-signal words with a short explanation. Here is a sample:

SIGHUP 1 Term Hangup detected on controlling terminal

or death of controlling process

SIGINT 2 Term Interrupt from keyboard

SIGQUIT 3 Core Quit from keyboard

SIGILL 4 Core Illegal Instruction

SIGABRT 6 Core Abort signal from abort(3)

SIGFPE 8 Core Floating point exception

SIGKILL 9 Term Kill signal

SIGSEGV 11 Core Invalid memory reference

# Torque

SIGPIPE 13 Term Broken pipe: write to pipe with no readers

SIGALRM 14 Term Timer signal from alarm(2)

SIGTERM 15 Term Termination signal

Some Linux systems set a bit in the upper byte of a 16-bit exit code when a process is terminated by a signal. Removing the bit (exit code modulo 256) yields the signal number. To calculate for the reported problem,  $265 \text{ modulo } 256 = 9$ , which is the SIGKILL signal, meaning the process was killed by someone, which happens to be the one common cause in all three items in the list of suspicious non-coincidences above.

Normally, the exit code is stored in an 8-bit unsigned integer, at least by shell programs, but a C program can retrieve the exit code from Linux as an integer with more than 8 bits. Obviously, this is the case with TORQUE's pbs\_mom, which is probably why users are confused since they likely believe an exit code can only have 0-255 as a value. However, this is not always true.

Below is a layout of the two 8-bit bytes in a 16-bit integer that has the value 265. You can see the upper byte (15-08) has the value "00000001" (1) representing the bit that indicates the exit code contains a signal number and the lower byte (07-00) has the value "00001001" (9), which is the signal number. Interpreted as a single integer, the value "0000000100001001" is 265, which is what TORQUE and Moab are storing and reporting.

|15-14-13-12-11-10-09-08|07-06-05-04-03-02-01-00|

| 0 0 0 0 0 0 0 1 | 0 0 0 0 1 0 0 0 |

|15-14-13-12-11-10-09-08|07-06-05-04-03-02-01-00|

So, an exit code of 265 is not an error. This is simply how some Unix and Linux operating system distributions work.

Other Unix and Linux variants indicate a signal in the exit code by setting the most-significant bit in the 8-bit exit code byte, which means signal 9 would be "10001001" (shown below), which is 137. This exit code value has been seen by other customers in similar situations (killing the job). Doing the same modulo arithmetic, except with 128 instead of 256, yields  $137 \text{ modulo } 128 = 9$ , which is signal 9 or SIGKILL.

|07-06-05-04-03-02-01-00|

| 1 0 0 0 1 0 0 0 |

# Torque

|07-06-05-04-03-02-01-00|

Unique solution ID: #1017

Author: Jason Booth

Last update: 2017-02-24 01:44