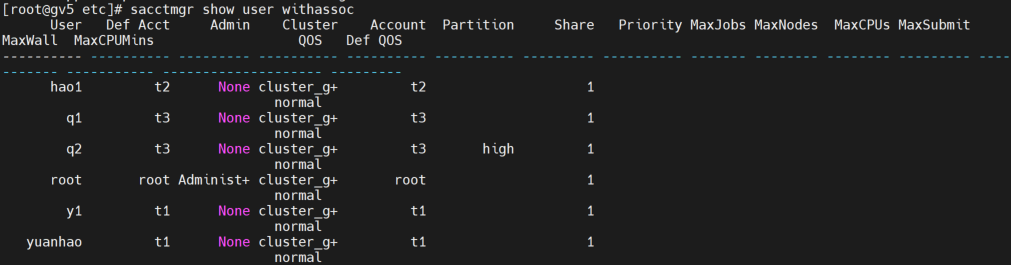
# Qos调研1222

## 用户优先级

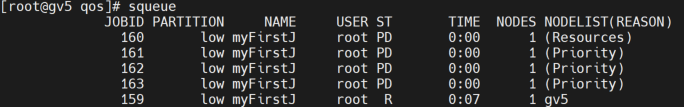


对这部分暂未做测试

Priority

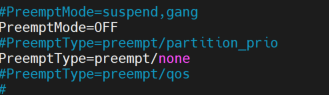
What priority will be added to a job's priority when using this association. This is overridden if set directly on a user. Default is the cluster's limit. To clear a previously set value use the modify command with a new value of -1.

## 未配置抢占



## 添加两个qos配置队列内抢占

### 未配置qos抢占规则



sacctmgr add qos test1

sacctmgr add qos test2

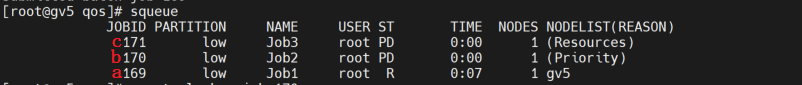
#修改用户qos中的优先级

sacctmgr modify qos normal set Priority=5

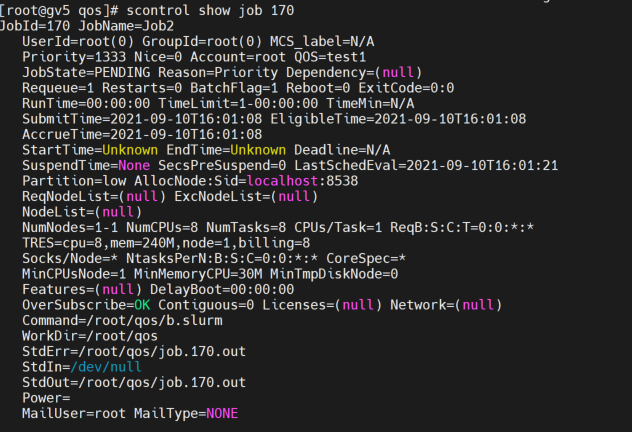
sacctmgr modify qos test1 set Priority=10

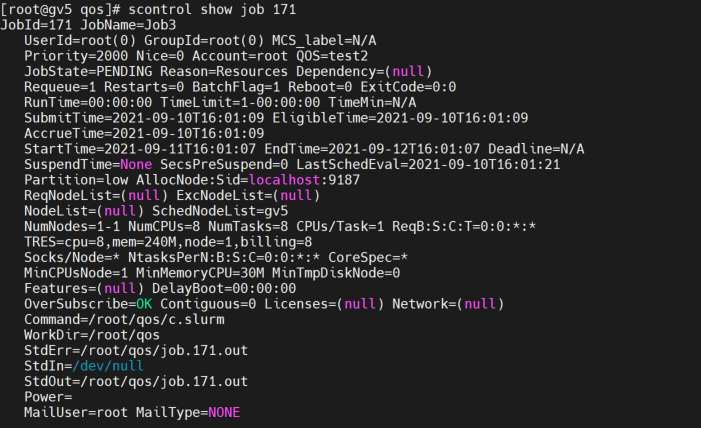
sacctmgr modify qos test2 set Priority=30

提交三个作业a,b,c



通过对作业状态查看b作业的priority为1333，c作业的priority：2000，所以排队顺序发生改变

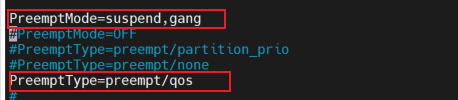




### 配置qos抢占规则

|  |  |
| --- | --- |
| Slurm抢占关系配置项 | 选项 |
| PreemptType | preempt/none #不开启抢占关系  preempt/partition\_prio #设置为分区抢占  preempt/qos #设置为qos抢占 |
| PreemptMode | OFF #不发生抢占  CANCEL #被抢占作业关闭  GANG #抢占调度器  REQUEUE #被抢占作业重新排队  SUSPEND #被抢占作业挂起 |

**注意：**与GANG结合使用，如果配置了PreemptType=preempt/qos，并且被抢占的作业和来自同一个队列的抢占作业在同一个队列上，那么它们将与Gang调度器共享资源(时间切片)。如果不是(即，如果被抢占者和被抢占者在不同的队列上)，那么被抢占的作业将保持挂起状态，直到被抢占者结束。



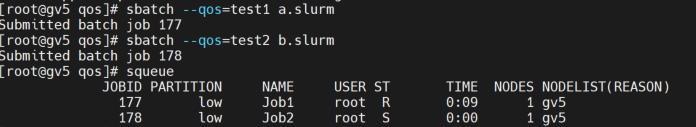
两种情况：QOS 的优先级与 QOS 抢占无关，仅使用 Preempt 来定义哪些 QOS 可以抢占其他 QOS。

首先设置test2可以抢占test1

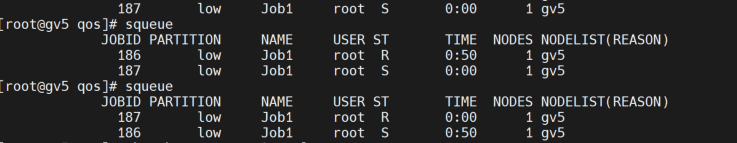
sacctmgr modify qos where name=test2 set Preempt=test1

sbatch --qos=test1 a.slurm

sbatch --qos=test2 b.slurm



现象：186首先运行，一段时间后发生抢占，187开始运行，且配置抢占后，186和187交替挂起。



时间片（SchedulerTimeSlice）结束后再次发生状态切换

结论：同一用户发生抢占配置规则：

1. PreemptType设置为preempt/qos，作业抢占规则由 Slurm 数据库中的服务质量 (QOS) 规范指定。

sacctmgr modify qos where name=test2 set Preempt=test1

此选项与PreemptMode=OFF不兼容。PreemptMode=SUSPEND 的配置 仅由 SelectType=select/cons\_res和SelectType=select/cons\_tres 插件支持。

1. PreemptMode：用于抢占作业或启用Gang调度的机制。此外，如果PreemptType=preempt/partition\_prio ，可以在每个分区的基础上指定

PreemptMode以覆盖此默认值。或者，如果PreemptType=preempt/qos，它可以在每个 QOS 的基础上指定。

1. GANG选项用于启用独立于是否启用抢占（即独立于PreemptType）的帮派调度，启用同一分区中作业的gang调度（时间切片），并允许恢复挂起的作业。
2. 注意：由于 gang 调度是针对每个分区独立执行的，如果使用PreemptType=preempt/partition\_prio，则较高 PriorityTier 分区中的作业将挂起较低 PriorityTier 分区中的作业以在释放的资源上运行。只有当抢占者作业结束时，Gang 调度程序才会恢复暂停的作业。如果配置了PreemptType=preempt/qos并且如果被抢占的作业和来自的抢占作业在同一个分区上，那么它们将与 Gang 调度程序共享资源（时间分片）。如果不是（即如果抢占者和抢占者在不同的分区上），则被抢占的作业将保持挂起，直到抢占者结束。
3. OverSubscribe：控制分区在每个资源（节点、套接字或核心，取决于 SelectTypeParameters 的值）上一次执行多个作业的能力。 如果要超额订阅资源，避免内存超额订阅非常重要。 SelectTypeParameters 应配置为将内存视为可消耗资源，并且 --mem 选项应用于作业分配。 资源共享通常仅在使用群调度时有用（PreemptMode=suspend,gang）。 OverSubscribe 的可能值是“EXCLUSIVE”、“FORCE”、“YES”和“NO”。 请注意，“YES”或“FORCE”的值会对具有数千个正在运行的作业的系统的性能产生负面影响。 默认值为NO，表示该资源只能运行一个资源，不允许抢占。 对于使用suspend/resume机制的作业抢占的所有分区，将分区的OverSubscribe设置配置为 FORCE。该FORCE选项支持一个额外的参数控制有多少职位能够超额认购一个计算资源（FORCE [：max\_share]）。默认情况下，max\_share 值为 4。为了抢占作业（而不是Gang调度），始终将 max\_share 设置为 1。要允许该分区中最多 2 个作业分配给公共资源（和Gang调度），请设置 FORCE:2。注意： PreemptType=preempt/qos如果由于作业抢占而启动，将允许在分区上运行一个额外的作业。例如，OverSubscribe=FORCE:1的配置通常每个资源只允许一个作业，但如果通过基于 QOS 的抢占来启动第二个作业。

# 队列之间抢占

## 相同qos之间抢占

sbatch -p low a.slurm

sbatch -p middle a.slurm

默认qos为normal

此时，高优先级队列直接抢占低优先级，



## 不同qos之间抢占

slurm20.02版本出现问题：

1. 情况

PreemptMode=suspend,gang

#PreemptMode=OFF

PreemptType=preempt/partition\_prio

在slurm.conf的设置为上面时，低优先级分区不会抢占高优先级分区，作业状态为正常排队

1. 情况2

PreemptMode=suspend,gang

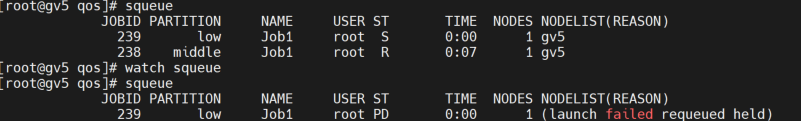
#PreemptMode=OFF

#PreemptType=preempt/partition\_prio

#PreemptType=preempt/none

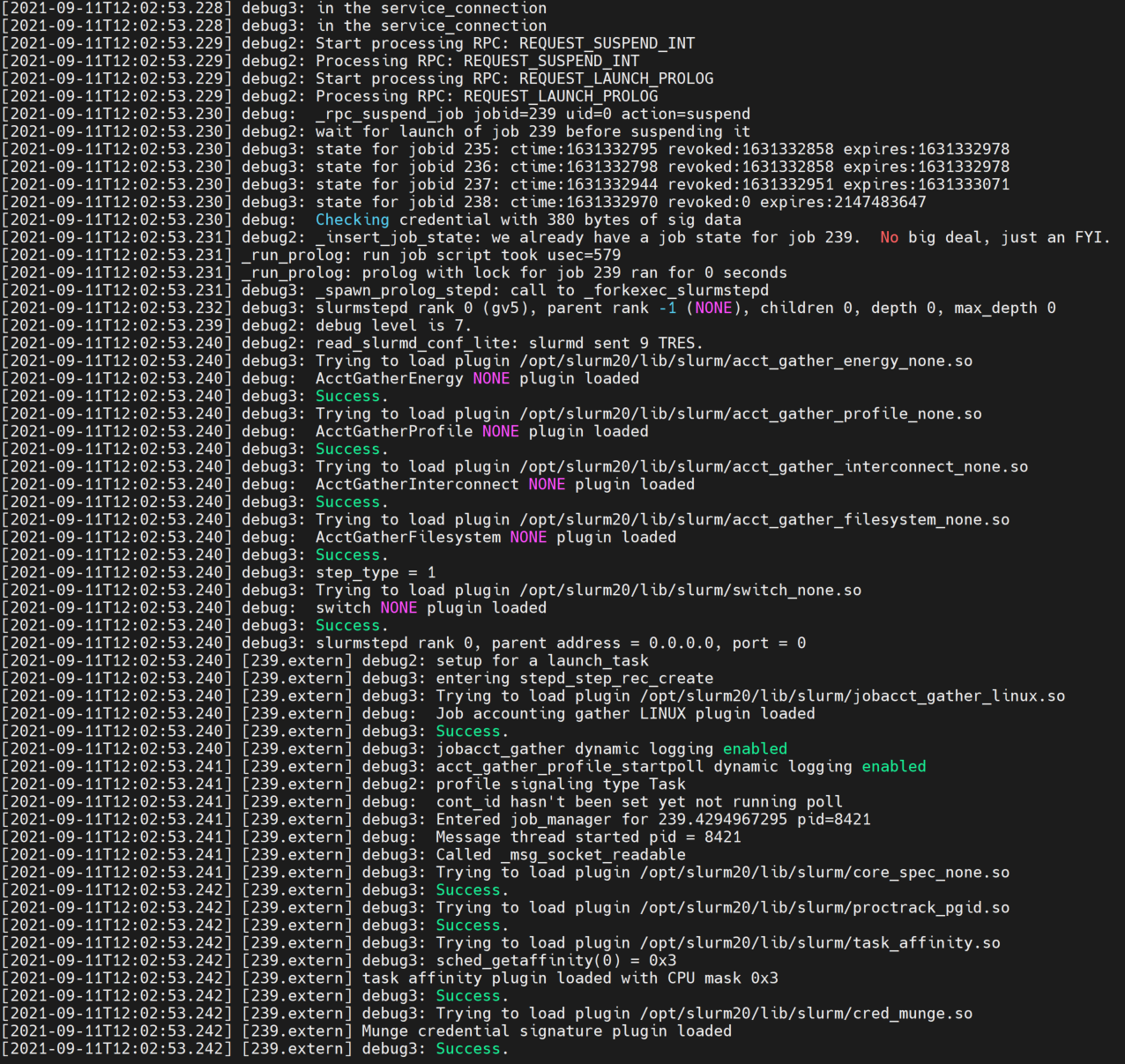
PreemptType=preempt/qos

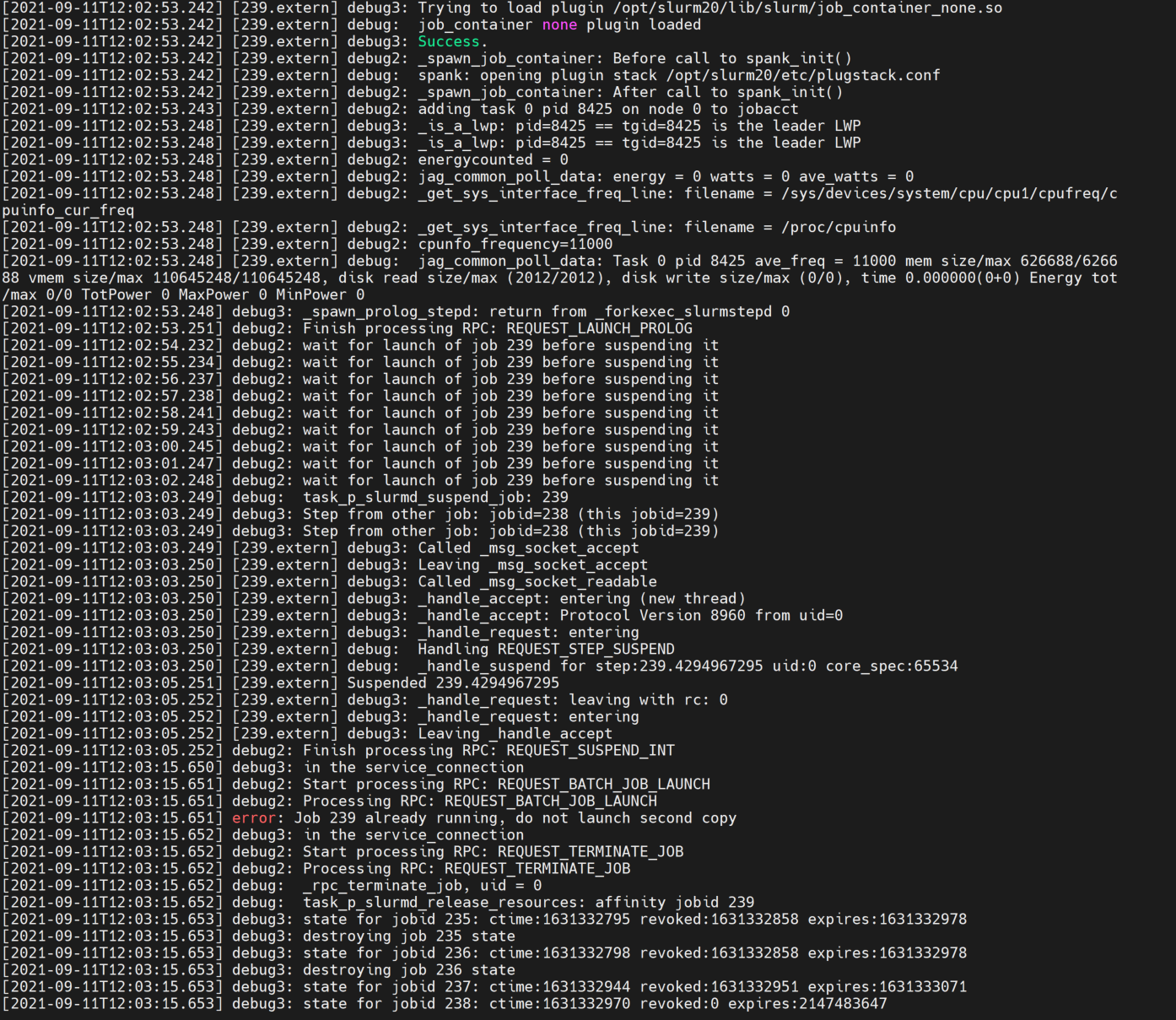
配置为上述情况时，低优先级分区会抢占高优先级分区，当低优先级作业提交时会被直接挂起，但是一段时间后会出现

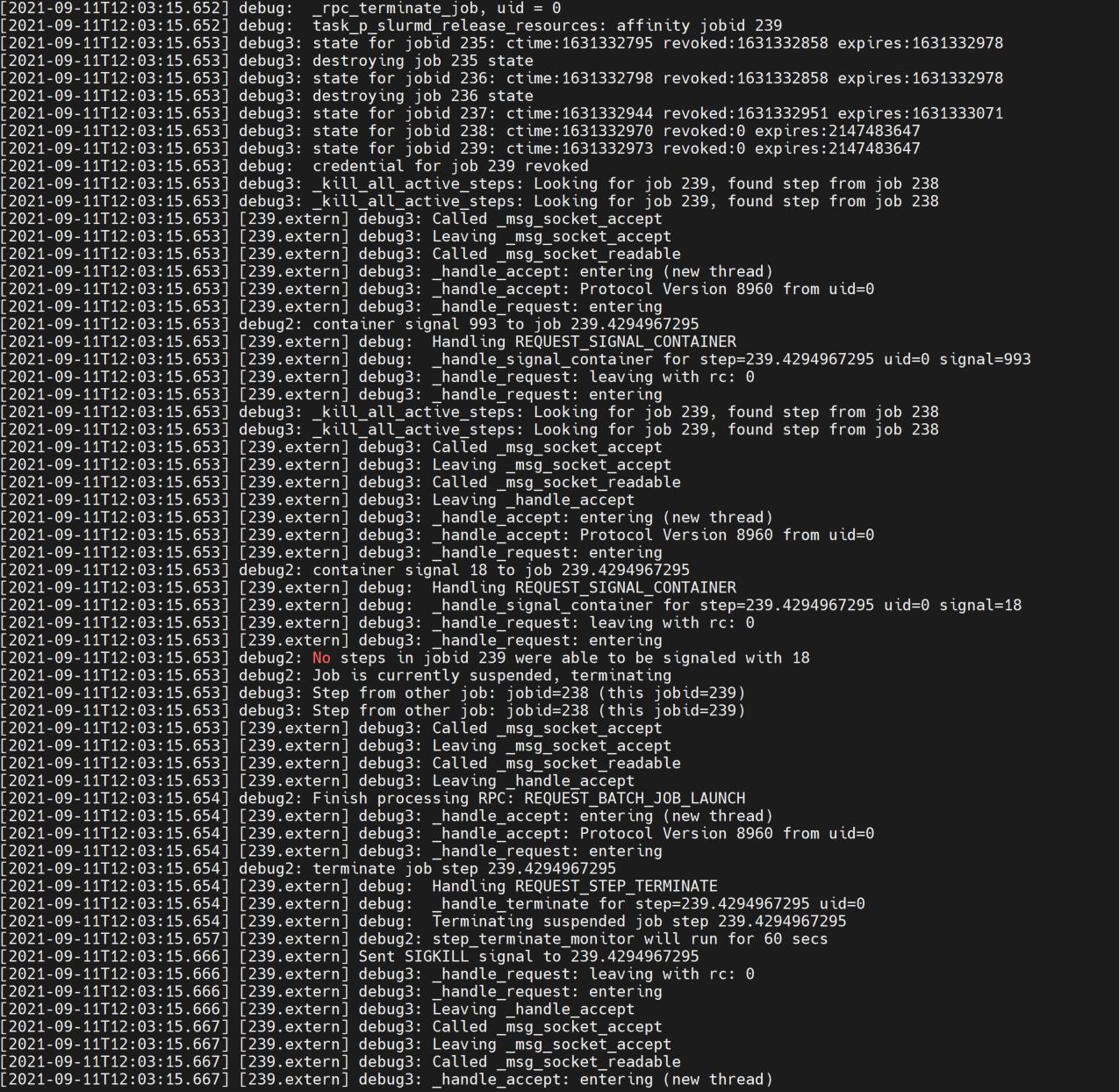


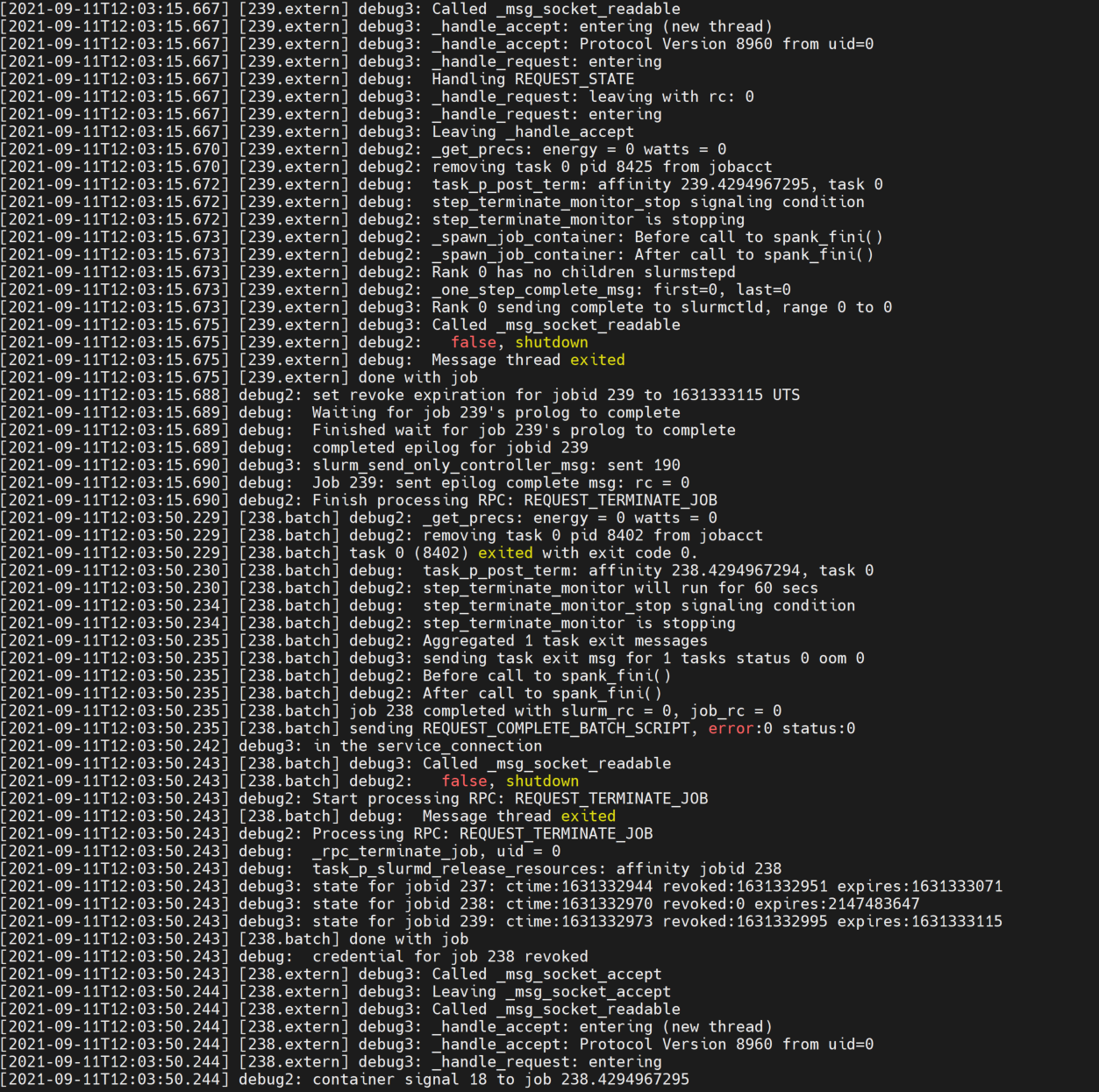
并且此时计算节点会处于drain状态，必须进行resume

slurmd日志









1. 高优先级分区抢占低优先级分区可以抢占，但是被挂起的作业会和抢占作业同时结束。此时squeue查询不到正在运行的作业