

安徽大学 HPC 平台资源管理优化策略

文 / 段运生

随着信息技术的飞速发展，高性能计算 HPC 得到众多科研领域的青睐，如生物基因、材料设计、金融计算、计算机科学等领域^[1]。高性能计算是计算机科学的分支之一，其采用并行化技术，将计算复杂度较高的计算任务分解成众多小型计算任务，将其分配给众多处理器同时计算，以实现高性能计算的目的^[2]。目前，高性能计算已成为继理论科学和实验科学之后科学探索的第三范式，被广泛应用在高能物理学、材料科学、航天航空飞行器设计、国民经济预测与决策、能源勘探、卫星图像处理、情报分析、互联网服务、工业仿真等领域，对国民经济发展和国防建设具有重要价值。

高性能计算平台一般由管理节点、登录节点、网络管理节点、计算节点和存储资源构成，其中计算节点一般包括CPU计算节点和GPU计算节点^[3]。此外，高性能计算平台还可以扩展公有云的校外计算资源，由此构成一套完整的异构化的混合计算平台。高性能计算平台中各计算节点由高性能网络（InfiniBand/ OPA）相互连接，构成一个高性能计算集群。

Slurm 是可用于高性能计算集群的开源集群管理器和作业调度系统，维护待处理的作业队列并管理此集群的整体资源^[4,5]。针对 Slurm 的调度软件的应用，众多学者在其基础上提出了许多适应于自身平台的优化改进策略。例如，为应对超大规模计算系统带来的大量监控数据，文献 [6] 提出了超大规模计算系统的监控、调度及网络优化实践，使集群节点

数量可以达到上万。针对 Slurm 的计费功能，文献 [7] 提出了一种基于 Slurm 作业调度系统的预计费装置和方法，通过将 Slurm 调度系统与 GOLD 机时记账系统集成，将作业使用机时从用户可用机时中预约，作业结束后扣除机时。此外，文献 [8] 还提

出一种基于 Slurm 作业管理的可视化调度系统，通过 Web 前端实现了用户以可视化方式使用作业调度系统。

Slurm 系统也存在不足，Slurm 调度系统默认实现用户在各节点间的无障碍跳转，也就是说用户可以通过登录节点直接 SSH 登录任意计算节点，并在此节点直接绕过 Slurm 调度软件进行作业计算^[9]。这将导致用户可以不再受 Slurm 资源管理的限制，随意使用计算资源，造成管理和运维的不便。此外，Slurm 在登录节点的监管方面也存在严重缺陷，用户可以直接无限制使用资源，而这将增加平台登录节点宕机的可能性。

针对上述问题，本文提出一套基于Slurm的智能化高性能计算资源管理方法，该方法包含一系列的控制功能：限制节点访问、智能化管理资源、资源异常查杀等，为平台的有效运行提供了一套完备的管理手段和策略。

Slurm 简介

Slurm 资源调度工具是面向 Linux、Unix 类似内核的免费和开源工作调度程

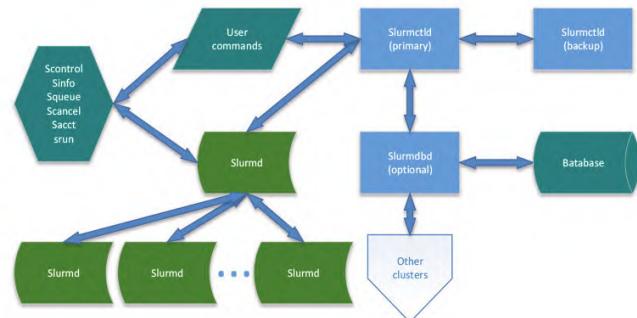


图 1 Slurm 系统结构

序，由世界上许多超级计算机和计算机集群使用。它提供了三个关键功能：首先，它在一段时间内为用户分配对资源（计算机节点）独占和 / 或非独占的访问权限，以便他们可以执行工作；其次，它提供了一个框架，用于在一组分配的节点上启动、执行和监视工作（通常是并行作业，例如 MPI）；最后，它通过管理待处理作业队列来仲裁资源争用。

Slurm 的设计非常模块化，有大约 100 个可选插件。在最简单的配置中，它可以在几分钟内完成安装和配置，并为更复杂的配置提供数据库集成，管理资源限制和工作负载优先级^[10]。

Slurm 资源管理系统的正常运行还需要一些系统服务与支撑环境的正确设置，比如系统配置、节点、分区、调度、记账存储、网络拓扑等。Slurm 提供了丰富的配置文件，如主配置文件、记账存储服务配置文件、节点配置文件、分区配置文件和通用资源配置文件等。

Slurm 资源管理系统由多个部分构成，如控制进程、记账存储进程、节点监控进程、作业管理进程、命令工具等，其系统结构如图 1 所示。

基于 Slurm 的资源管理策略优化

本文主要针对高性能计算集群的登录节点、管理节点和计算节点进行资源和权限的管理配置，其中使用到的软件资源有：Slurm、CpuLimit 模块、pam_access.so 插件和 Linux 基本的指令集。实现的功能如下：

1. 对各节点的 root 进行固定 IP 访问限制；
2. 限制登录节点中用户的资源使用；
3. 限制用户登录无作业的计算节点；
4. 在计算节点查杀非 Slurm 用户的大计算进程；
5. GPU 节点的异常作业处理。

通过上述工具配置相应策略，可实现高性能计算平台的安全运行和资源智能化、自动化的监管与调度。

root 固定 IP 访问限制

高性能计算平台默认为用户提供计算服务器，所以访问端口对外开放。root 用户拥有整个系统中至高无上的权力，为了避免一些潜在的风险，我们需对 root 用户进行访问权限的设置。该功能需要使用 pam_access.so 插件，该插件的功能和作用是根据主机名（包括普通主机名或者 FQDN）、IP 地址和用户实现全面的访问控制。pam_access.so 模块的具体工作行为根据配置文件 /etc/security/access.conf 来决定。具体设置策略如下：

管理节点只允许指定 IP 地址 root 访问，禁止非 root 用户访问；登录节点允许指定 IP 的 root 访问和所有用户访问；计算节点允许指定 IP 的 root 访问，禁止用户直接访问。

实现方法：

1. 在 /etc/pam.d/sshd 添加如下一句
account required pam_access.so
2. 在 /etc/security/access.conf 添加
+ : root : xxx.xxx.xxx.xxx
- : root : ALL
+ : ALL : ALL

该配置文件的主体包含了三个字段：权限、用户和访问发起方。格式上是一个用 “:” 隔开的表。其中第一个字段：权

限（permission），使用 “+” 表示授予权限，用 “-” 表示禁止权限。第二个字段：用户（user），定义了用户、组以及用 “@” 表示的不同主机上的同名用户和同一主机上不同名用户。第三个字段：访问发起方（origins），定义了发起访问的主机名称、域名称和终端名称。

登录节点中用户的资源使用限制

登录节点是高性能计算的门户，主要用于用户的登录访问和作业提交功能。若用户在登录节点执行大规模的计算任务，会导致该登录节点宕机。我们通过定时检测用户的进程信息，对 CPU 使用率比较高的进程进行 CPU 资源限制。CPU 资源限制选用的是 cpulimit 插件，具体设置策略如下：

1. 通过在登录节点执行如下命令实现对用户进程的获取（grep -h "[0-9][4]" 是依据安徽大学的账户规则进行的匹配筛选）：

```
ps -o ruser=userForLongName -e -o pid,%cpu
--sort=-%cpu |grep -v root |grep -v PID|head
-10 |grep -h '^.[0-9][4]'
```

2. 对检测到的进行 pid，进行 CPU 资源对比，超过阈值的 pid 进行 cpu 限制：

```
if [ $(echo "$cpu>80" |bc) == 1 ]; then
cpulimit --pid $pid --limit 50 &
fi
```

登录节点的大规模计算资源限制的处理流程如图 2 所示，配置 crontab 定时任务计划，使脚本定制执行。

限制用户登录无作业的计算节点

该功能主要实现用户在登录节点无法直接 SSH 到计算节点，只有用户通过

Slurm 提交作业后，才有权限 SSH 到运行该作业的计算节点。本功能需要使用 Slurm 的 PAM 插件来实现。

实现方法：

1. 添加不包含 pam_systemd.so 的 PAM 配置：

```
grep -v pam_systemd.so /etc/pam.d/password-
auth> /etc/pam.d/password-auth-no-systemd
2./etc/security/access.conf 里添加如下:
```

```
+:root:ALL
-:ALL:ALL
```

3./etc/pam.d/sshd 添加：
account sufficient pam_access.so
account required pam_slurm.so

4. 在 /etc/pam.d/ 目录下创建 slurm 文件，文件内容如下：

```
auth required pam_localuser.so
account required pam_unix.so
session required pam_limits.so
```

计算节点查杀非 Slurm 用户的大计算进程

用户在作业运行期间，可以 SSH 到相应的计算节点。这时用户可以直接在该计算节点进行提交作业操作，这个操作是系统不允许的。但是如果作业结束后，直接阻截该用户的所有进程，也不合理。为此，我们添加了对计算节点异常进程的查杀操作，操作流程如图 3 所示，其具体设置策略如下：

1. 通过 squeue 查询当前计算节点的作业信息；

2. 通过 ps 指令查询排除提交作业用户的进程信息；

3. 对各进程信息的 CPU 使用率进行

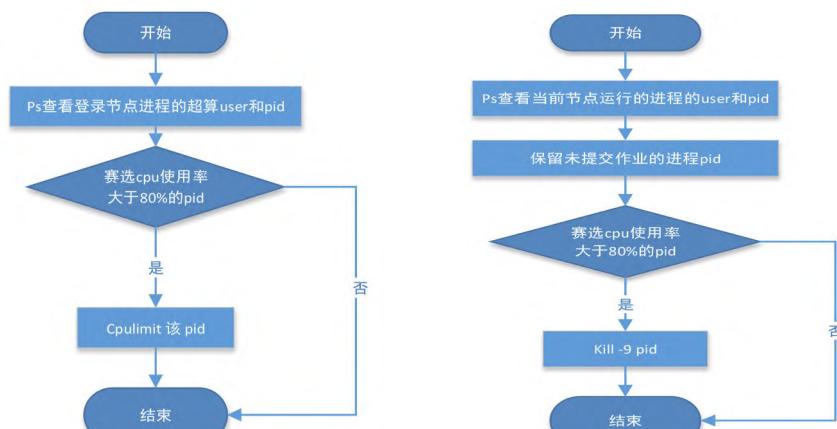


图 2 登录节点资源限制流程

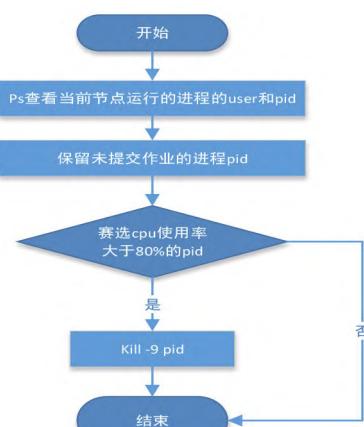


图 3 计算节点的异常进程处理流程

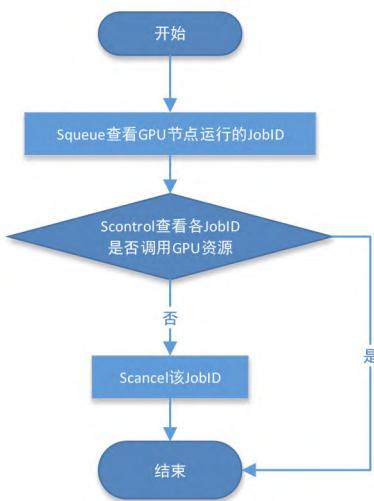


图 4 GPU 节点的异常进程处理流程

检测,对使用率较高的进程进行查杀操作。

GPU 节点异常作业处理

GPU 计算节点主要用于计算深度学习等高性能计算,不允许提交传统的 CPU 计算作业,但是 Slurm 无法准确判断和管理该功能。为此,我们编写了在异常 GPU 作业查杀的模块。其操作流程如图 4 所示,具体设置策略如下所示:

1. 通过 squeue 查询当前计算节点的作业信息;
2. 通过 scontrol 轮询各作业,判断是否有申请 GPU 的字段信息(TresPerNode=gpu),对未使用 GPU 的作业进行 scancel 操作。

策略部署与性能测试

本文针对安徽大学校级高性能计算平台在实际运行中遇到的问题提出了基于 Slurm 的资源管理调度优化方法。该优化策略包括:根据 IP 地址的类型确定是否允许 IP 地址访问对应的节点,以及根据用户的类型确定是否允许用户访问对应的节点;根据用户在登录节点使用的 CPU 资源和预设的 CPU 资源阈值,确定是否进行 CPU 限制;根据用户提交作业的情况,确定是否允许用户登录到计算节点;获取已提交作业的用户信息,根据

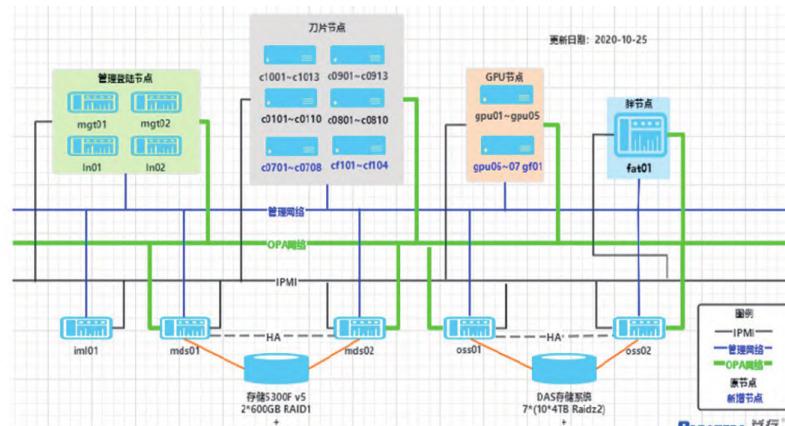


图 5 安徽大学高性能计算平台拓扑

非作业用户进程的 CPU 使用率和预设的使用率阈值,确定是否查杀该用户进程;对 GPU 计算节点中的非 GPU 作业进行取消等功能。

本文将 Slurm 的优化策略在安徽大学校级高性能计算平台进行部署,并测试其性能。安徽大学高性能计算平台是集传统超算和 GPU 计算于一体的混合计算平台,该平台系统拓扑图如图 5 所示。

目前,该平台的建设规模如下:

存储空间 186T, CPU 总核数 3340 核, GPU 总卡数 59 张, 总计算节点数量为 85 个 CPU 节点和 10 个 GPU 节点, 平台单精度性能总算力 1316.8 Tflops。

安徽大学高性能计算平台于 2021 年底部署了基于 Slurm 的资源管理优化策略,并结合用户培训、群里通知等方式协助管理。经过 2022 年第一季度实际测试,统计异常平台记录如表 1 所示。从表中可以看出,平台的异常 IP 的节点访问现象显著降低,有效解决了登录节点异常宕机的问题,同时切实保证了作业正常运行,极大减少了平台运维工程师的工作量。

表 1 安徽大学高性能计算平台的异常信息统计

序号	2022.01	2022.02	2022.03	2022.04
IP 异常登录次数	53	21	12	8
登录节点异常作业次数	159	43	13	6
异常 CPU 作业次数	21	8	3	4
GPU 节点的异作业次数	32	11	6	2

本文提出一种基于 Slurm 的智能化的高性能计算平台的资源管理优化策略,其能够保证高性能计算平台资源管理和分配,有效避免了 Slurm 的常见漏洞;此外,该策略能够限制 root 的安全访问权限和计算节点的权限访问,同时也限制登录节点中用户的资源使用和计算节点的异常进程查杀。CEN (责编:陈荣)

(作者单位为安徽大学网络信息中心)

基金项目:安徽省教育厅高校自然科学重点项目(KJ2021A0013)。

参考文献:

- [1] 聂含伊, 杨希, 张文桔. 面向多领域的高性能计算机应用综述 [J]. 计算机工程与科学, 2018, 40(1): 1–9.
- [2] 汪洋, 李鹏, 季一木, 等. 高性能计算与天文大数据研究综述 [J]. 计算机科学, 2020, 47(1): 1–6.
- [3] 唐少宇. 高性能计算机体系结构综述 [J]. 计算机光盘软件与应用, 2010, 15: 51–52.
- [4] 郑文旭, 潘晓东, 马迪, 等. 用于高性能计算的作业调度能效性研究综述 [J]. 计算机工程与科学, 2019, 041(009): 1525–1533.
- [5] Bacchini F, Olshevsky V, Lapenta G. Slurm: An innovative Particle-in-Cell Method for Magnetohydrodynamics [J]. American Physical Society, 2016: 1–1.
- [6] 秦晓宁, 王家尧, 胡梦龙, 等. 面向超大规模计算系统的监控、调度及网络优化实践 [J]. 数据与计算发展前沿, 2020, 2(1): 1–5.
- [7] 郝文静, 张涛, 原帅, 等. 一种基于 SLURM 作业调度系统的预计费装置和方法 [P]. CN111198755A. 2020.
- [8] 曲德源, 吴登勇, 孙超, 等. 一种基于 Slurm 作业管理的可视化调度系统 [P]. CN108319514A. 2018.
- [9] 杨敏, 李淑倩, 关宇, 等. Slurm 作业调度系统在高性能集群中的应用 [J]. 电脑知识与技术: 学术版, 2021, 17(36): 1–3.
- [10] 张威. 基于超级计算的复杂 CAE 作业管理系统的研发与实现 [D]. 湖南大学, 2015.