

HPC 接口文档（平台）

签名

- url: /api/sign
- 请求方式: POST
- 请求参数:
 - cipher: 密钥
- 返回示例

```
{
  code:200
  data:{}
  msg:"签名成功"
}
```

一. HPC 资源

1.1 队列

队列接口为 POST 的请求参数，兼容 slurm 所有字段，字段及解释请参考 slurm 官方文档 <https://slurm.schedmd.com/scontrol.html#lBAK>

1.1.1 新增队列

- url: /partiton/createPartition
- 请求方式: POST
- 请求参数: {}

- 返回示例

```
{  
  code:200  
  data:""  
  msg:"scontrol create PartitionName=test AllocNodes=n[100-200] 执行成功"  
}
```

1.1.2 删除队列

- URL: /partiton/deletePartition
- 请求方式: POST
- 请求参数: {}
- 返回示例

```
{  
  code:200  
  data:""  
  msg:"scontrol delete PartitionName=test AllocNodes=n[100-200] 执行成功"  
}
```

1.1.3 修改队列

- URL: /partiton/updatePartition
- 请求方式: POST
- 请求参数: {}
- 返回示例

```
{  
  code:200  
  data:""
```

```
msg:"scontrol update PartitionName=test AllocNodes=n[300-400] 执行成功"
}
```

1.1.4 显示队列

- URL: /partiton/showPartition
- 请求方式: GET
- 请求参数:
 - partitionName
- 返回示例

```
{
code:200
data:"PartitionName=debug
AllocNodes=ALL AllowGroups=ALL Default=YES
DefaultTime=NONE DisableRootJobs=NO Hidden=NO
MaxNodes=UNLIMITED MaxTime=UNLIMITED MinNodes=1
Nodes=snowflake[0-48]
Priority=1 RootOnly=NO OverSubscribe=YES:4
State=UP TotalCPUs=694 TotalNodes=49"
msg:"scontrol show PartitionName=debug 执行成功"
}
```

1.2 集群账号

1.2.1 新增账号

- url: /ldapAccount/addLdapAccount
- 请求方式: POST
- 请求参数: LdapAccount

- 返回示例

```
{  
  code:200  
  data:""  
  msg:""  
}
```

1.2.2 删除账号

- url: /ldapAccount/deleteLdapAccount
- 请求方式: POST
- 请求参数: LdapAccount
- 返回示例

```
{  
  code:200  
  data:""  
  msg:""  
}
```

1.2.3 修改账号

- url: /ldapAccount/editLdapAccount
- 请求方式: POST
- 请求参数: LdapAccount
- 返回示例

```
{  
  code:200  
  data:""
```

```
msg:""
```

1.2.4 查询账号

- url: /ldapAccount/searchLdapAccount
- 请求方式: POST
- 请求参数: LdapAccount
- 返回示例

```
{
  code:200
  data:""
  msg:""
}
```

1.3 存储

1.3.1 查询存储

- url: storage/queryStorage
- 请求方式: GET
- 请求参数:
 - mntPoint: 挂载点
 - username: 用户名
- 返回示例

```
{
  code:200
```

```
data: "/sh3/xxxx/home/yeesuan123456
      1849907988      0 2048000000      - 5591423      0      0      -
uid 200000 is using default file quota setting"
msg: ""
}
```

1.3.2 修改存储

- url: storage/editStorage
- 请求方式: POST
- 请求参数:

```
{
username: "", //要限制的用户
mntPoint: "", //挂载点
limit: "", //限制大小
}
```

- 返回示例

```
{
code: 200
data: ""
msg: ""
}
```

1.4. 节点

1.4.1 查询

- url: nodes/showNodes
- 请求方式: GET

- 请求参数
 - nodes: 节点号,可为空
- 返回示例

```
{
code:200
data:"NodeName=n07849 Arch=x86_64 CoresPerSocket=28
CPUAlloc=0 CPUTot=56 CPULoad=0.03
AvailableFeatures=(null)
ActiveFeatures=(null)
Gres=(null)
NodeAddr=n07849 NodeHostName=n07849 Version=19.05.7
OS=Linux 3.10.0-957.el7.x86_64 #1 SMP Thu Nov 8 23:39:32 UTC 2018
RealMemory=384000 AllocMem=0 FreeMem=377846 Sockets=2 Boards=1
State=IDLE ThreadsPerCore=1 TmpDisk=0 Weight=1 Owner=N/A MCS_label=N/A
Partitions=q_ys_test3,q_ysuan_384,yeesuan
BootTime=2022-05-29T17:08:12 SlurmdStartTime=2022-12-09T11:07:03
CfgTRES=cpu=56,mem=375G,billing=56
AllocTRES=
CapWatts=n/a
CurrentWatts=0 AveWatts=0
ExtSensorsJoules=n/s ExtSensorsWatts=0 ExtSensorsTemp=n/s"
msg:""
}
```

二. 文件模块

2.1 登录 SFTP

- url: sftp/v1/login
- 请求方式:POST

- 请求头: Header 携带"Authorization"
- 请求参数

```
{
  "id": "用户 ID",
  "plat_account": "平台账号",
  "typeSpec": "账号类型默认为 yeesuan",
  "ldapAccountId": "系统账号 ID"
}
```

- 返回示例:

后续所有请求均在 url 中增加 uid=connId, connId 为 Login 请求成功返回的信息

```
{
  "data": "connId",
  "message": "*****",
  "status": true
}
```

2.2 获取当前目录的文件列表

- url: /sftp/v1/get_dirs
- 请求方式: GET
- 请求头: Header 携带"Authorization"
- 请求参数
 - path
- 返回示例:


```
{
  "data": {
    "file_info_list": [
      {
        "file_name": ".TopSAP-3.5.2.26.2-x86_64.bin.dir",
        "file_type": "d", //文件类型（d:表示目录文件， -:表示普通文件，
                          L-:表示普通链接文件， Ld:表示目录连接文件）
        "file_size": 30,
        "file_perm": "-rwxr-xr-x", //文件权限
        "file_user": "0 0", //
        "file_time": "2021-11-17 11:24:24" //文件创建时间
      }
    ],
    "current_path": "/root" //绝对路径
  },
  "message": "获取文件列表成功!",
  "status": true
}
```

2.3 上传文件

- url: /sftp/v1/upload
- 请求方式:POST
- 请求参数:
 - file: 二进制切片文件
 - file_count: 切片总总数
 - file: 二进制切片文件
 - file_count: 切片总总数
 - file_index: 切片序号

- file_length: 当前切片文件的大小（单位为字节）
- start_byte: 当前切片在总文件的起始位
- path: 需要上传到的路径（最后不能加'/'）
- total_length: 总文件大小(单位为字节)
- file_name: 文件名
- file_id: 文件 ID（文件大小+文件名）
- md_5_sum: 当前切片的哈希值
- 返回示例:

```
{
  "data": "",
  "message": "文件上传成功!",
  "status": true
}
```

2.4 文件确认

- url: /sftp/v1/confirm
- 请求方式: POST
- 请求参数类型: JSON 请求
- 请求参数:
 - file_count: 切片总总数
 - path: 需要上传到的路径（最后不能加'/'）
 - file_name: 文件名
 - file_id: 文件 ID（文件大小+文件名）

- 返回示例

```
{
  "status": true,
  "message": "文件确认成功!",
  "data": ""
}
```

2.5 下载某个文件

- url: /sftp/v1/download
- 请求方式: GET
- 请求 Cookie: Cookie: uid=wi2jd392jda32993ei;
- 请求参数:
 - path:文件所在路径
- 返回示例: 无

2.6 高速下载

- url: /sftp/v1/fast_download
- 请求方式: GET
- 请求参数:
 - path:文件路径
 - start:number

2.7 获取文件哈希值

- url: /sftp/v1/get_file_sum

- 请求方式: GET
- 请求参数:
 - path:文件路径

2.8 创建新的目录

- url: /sftp/v1/create_dir
- 请求方式: POST
- 请求参数类型: String
- 请求参数:
 - path=*
- 返回示例:

```
{
  "data": "",
  "message": "创建目录成功!",
  "status": true
}
```

2.9 删除文件或目录

- url: /sftp/v1/delete_file
- 请求方式: POST
- 请求参数
 - path:文件/目录路径
- 返回示例:

```
{
  "data": "",
  "message": "删除目录/文件成功!",
  "status": true
}
```

2.10 本地文件复制

- url: /sftp/v1/local_copy_file
- 请求方式: POST
- 请求参数类型: JSON
- 请求参数:

```
{
  "from_path": "",
  "to_path": ""
}
```

- 返回示例:

```
{
  "data": "",
  "message": "文件 Copy 成功!",
  "status": true
}
```

2.11 多 SFTP 连接文件互拷

- url: /sftp/v1/cluster_copy_file
- 请求方式: POST
- 请求参数:

```
{
  "from_path": "",
  "to_path": "",
  "to_uid": ""
}
```

- 返回示例:

```
{
  "data": "",
  "message": "文件 Copy 成功!",
  "status": true
}
```

2.12 获取远程文件传输，传输完成率和预计完成时间

- url: /sftp/v1/get_completed_size
- 请求方式: POST
- 请求参数: NULL
- 返回示例:

```
{
  "data": {
    "file_name": "test.txt", //文件名
    "completed_rate": 0.3, //为 float64 类型数据最大不超过 1
    "remain_time": 124.9, //预计剩余时间，为 float64 类型数据，单位为秒
    "start_time": "2023-04-01 12:00:03", //传输开始时间
    "from_path": "/root/test.txt", //源路径
    "to_path": "/mnt/text.txt", //目标路径
    "from_id": "129334564645", //源 ID
    "to_id": "12334564645", //目标 ID
    "status": "running", //传输状态
  }
}
```

```

"file_type_format": ".txt", //文件后缀
"file_type": "-"/-: 表示文件, d: 表示目录, Ld: 表示链接目录, L-: 表示链接文件
"message": "获取完成率成功!",
"status": true
}

```

2.13 文件或目录移动（重命名）

- url: /sftp/v1/get_completed_size
- 请求方式: POST
- 请求参数:

```

{
"old_name": "",
"new_name": "",
"file_path": ""
}

```

- 返回示例:

```

{
"data": "",
"message": "文件移动成功!",
"status": true
}

```

2.14 关闭 server 端的连接

- url: /sftp/v1/close
- 请求方式: POST
- 请求参数: NULL
- 返回示例

```
{
  "data": "",
  "message": "连接关闭成功!",
  "status": true
}
```

2.15 文件搜索

- url: /sftp/v1/search
- 请求方式: GET
- 请求参数:
 - path=*
- 返回示例:

```
{
  "data": {
    "file_info_list": [
      {
        "file_name": ".TopSAP-3.5.2.26.2-x86_64.bin.dir",
        "file_type": "d",
        "file_size": 30,
        "file_perm": "-rwxr-xr-x",
        "file_user": "0 0",
        "file_time": "2021-11-17 11:24:24 +0800 CST"
      }
    ],
    "current_path": "/root"
  },
  "message": "文件搜索成功!",
  "status": true
}
```


2.16 文件压缩或解压缩

- url: /sftp/v1/file_compression

注：文件压缩和解压操作需要在服务器中安装对应的压缩工具（tar、zip、unzip）

- 请求方式：POST
- 请求类型：json
- 请求参数：

```
{
  "work_path": "", //工作目录，执行文件压缩或解压操作的目录
  "from_path": "", //源文件与工作目录的相对路径
  "type": 0, //压缩或解压操作类型
  "to_path": "" //目标文件与工作目录的相对路径
}
```