极智算代理商API接口文档

1. 文档说明

本文档为极智算平台（www.jygpu.com）官方代理商专属开放API接口规范文档，面向极智算授权代理商、技术开发团队、系统集成商，用于实现代理商自有业务系统与极智算算力平台的深度对接、资源分销、产品管理、数据对账等全链路业务功能。

1.1 适用范围

本API适用于：

- 代理商自有平台 / 系统与极智算算力平台的系统对接、数据互通。
- 实时获取算力资源池、规格、库存、价格、节点分布等基础数据。
- 第三方平台通过API调用极智算的算力资源。
- 企业内部系统与极智算平台的对接。

1.2 术语定义

术语	解释
代理商	与极智算平台合作代理销售算力资源的合作伙伴
容器	轻量级的应用运行环境
实例	指容器的运行实例，包含GPU资源

2. 认证方式

API密钥认证

1. **获取API密钥**：代理商在极智算平台申请且审核通过后获取API密钥
2. **请求认证**：在请求头Header中携带获取的API Key

请求头示例：

```
X-Access-Key: <access_key>
X-Signature:  <secret_key>
```

3. API基础信息

3.1 API地址

- https://www.jygpu.com/api

3.2 请求方式

支持GET、POST、PUT、DELETE等方法

3.3 数据格式

- 请求体：Query、Json格式
- 响应体：Json格式

3.4 响应结构

正确示例：

```
{
  "code": 0,
  "data": {},
  "message": "success"
}
```

字段	类型	说明
code	Integer	响应码，0表示成功，其他表示失败
data	Object	响应数据
msg	String	响应消息

通用错误示例：

X-Access-Key: 为空或错误

```
{
    "code": 7,
    "data": {},
    "msg": "未提供认证信息" / "无效的访问密钥"
}
```

X-Signature: 为空或错误

```
{
    "code": 7,
    "data": {},
    "msg": "缺少密钥信息" / "密钥验证失败"
}
```

接口响应 code=0 时为成功，code=7 时接口返回失败

4. 核心接口

4.1 资源相关接口

4.1.1 获取镜像列表

接口地址：/agent/user/images/getImagesList

请求方式：GET

请求参数：无

响应示例：


```
{
  "code": 0,
  "data": {
    "list": [
      {
        "ID": 2,
        "CreatedAt": "2025-04-28T09:56:17.639+08:00",
        "UpdatedAt": "2026-03-19T10:42:29.964+08:00",
        "name": "jiyun/cuda-jupyter-ssh", // 镜像name
        "version": "latest", // 镜像version
        "registry": "registry.jygpu.com", // 镜像仓库地址
        "remarks": "Jiyun/cuda-jupyter-ssh",
        "buildTime": "2025-04-13T18:29:07+08:00",
        "ports": "[{\"name\":\"SSH\",\"port\":\"22\"},
{\"name\":\"JupyterLab\",\"port\":\"8888\"}]",
        "type": "jiyun",
        "description": "CUDA开发环境，带Jupyter和SSH服务",
        "logo":
"uploads/file/8dd2002413f1d374577a4ca015282082_20260319102904.png",
        "category": "base",
        "categoryID": 10,
        "arch": "x86" // 镜像架构（arm/x86）
      }
    ],
    "total": 1
  },
  "msg": "获取成功"
}
```


4.1.2 获取分类镜像

接口地址：/agent/user/images/byType

请求方式：GET（Query格式传参）

请求参数：

参数名	类型	必填	说明
imageType	String	是	镜像类型，jiyun（基础镜像）/models（AI模型镜像）

响应示例：


```
{
```

```
  "code": 0,
  "data": [
    {
      "ID": 3,
      "CreatedAt": "2025-04-28T09:59:41.545+08:00",
      "UpdatedAt": "2026-03-19T10:00:58.146+08:00",
      "name": "jiyun/stable-diffusion-sd-v1-4", // 镜像name
      "version": "v2", // 镜像version
      "registry": "registry.jygpu.com", // 镜像仓库地址
      "remarks": "stable-diffusion-sd-v1-4",
      "buildTime": "2025-04-13T18:29:07+08:00",
      "ports": "[{\"name\":\"web\",\"port\":\"7860\"},
{\"name\":\"SSH\",\"port\":\"22\"}]",
      "type": "models",
      "description": "文生图大模型",
      "logo":
"uploads/file/0592db1dc6d6e674523e9aa19b252689_20260319095952.png",
      "category": "文生图模型",
      "categoryID": null,
      "arch": "x86" // 镜像架构（arm/x86）
    }
  ],
  "msg": "获取成功"
}
```


4.1.3 获取代理商镜像提交任务

接口地址：/agent/user/products/getUserProductCommits

请求方式：GET（Query格式传参）

请求参数：

参数名	类型	必填	说明
status	String	否	pushing（推送中）/ failed（失败）。不传时默认只返回 pushing 与 failed 两种状态;

响应示例：


```
{
  "code": 0,
  "data": [
    {
      "ID": 57,
      "CreatedAt": "2026-09-16T17:46:20.736+08:00",
      "UpdatedAt": "2026-09-16T18:01:49.03+08:00",
      "name": "7c75683518181835245",
      "node": "222.212.85.181",
      "imageName": "17623786102-ima:latest",
      "status": "failed",
      "msg": "denied: adding 2.7 GiB of storage resource, which when updated to
current usage of 1.9 GiB will exceed the configured upper limit of 2.0 GiB.",
      "registryUrl": "https://registry.jygpu.com"
```

```
    }
  ],
  "msg": "成功"
}
```

4.1.4 创建当前代理商 Harbor 仓库

接口地址：/agent/user/images/createImageRepoForUser

请求方式：POST

请求参数：无

响应示例：

```
{
  "code": 0,
  "data": {
    "username": "18181835245",
    "password": "aB3xY9kQ2mZp",
    "project": "zhisuan18181835245"
  },
  "msg": "创建成功"
}
```

4.1.5 获取当前代理商 Harbor 仓库信息

接口地址：/agent/user/images/getMyRepoInfo

请求方式：GET

请求参数：无

响应示例：

```
{
  "code": 0,
  "data": {
    "ID": 10,
    "CreatedAt": "2026-01-09T16:28:50.961+08:00",
    "UpdatedAt": "2026-01-09T16:28:50.961+08:00",
    "username": "18181835245",
    "harborUrl": "https://registry.jygpu.com",
    "harborUsername": "18181835245",
    "harborPassword": "P23QJeXAbGG1",
    "email": "18181835245@example.com",
    "defaultProject": "zhisuan18181835245"
  },
  "msg": "成功"
}
```

4.1.6 获取当前代理商 Harbor 项目中的镜像列表

接口地址：/agent/user/images/getHarborProjectImages

请求方式：GET

请求参数：无

响应示例：

```
{
  "code": 0,
  "data": [
    {
      "name": "zhisuan18181835245/19130046910-image-baocun", // 镜像名称
      "tag_count": 1,
      "artifacts": [
        {
          "size": 1676975168, // 镜像大小
          "push_time": "2026-09-23T03:17:44.651Z",
          "tags": [
            {
              "name": "latest", // 镜像tag
              "push_time": "2026-09-23T03:17:44.707Z"
            }
          ]
        }
      ]
    }
  ],
  "msg": "成功"
}
```

4.1.7 获取当前代理商 Harbor 项目配额

接口地址：/agent/user/images/getHarborProjectQuota

请求方式：GET

请求参数：无

响应示例：

```
{
  "code": 0,
  "data": {
    "quota": {
      "hard": {
        "storage": 107374182400
      },
      "used": {
        "storage": 13361209640
      }
    },
    "repo_count": 12
  },
  "msg": "成功"
```

```
}
```

4.1.8 删除当前代理商 Harbor 仓库中的镜像

接口地址：/agent/user/images/deleteHarborImage

请求方式： POST

请求参数：

参数名	类型	必填	说明
repository	String	是	仓库名，如zhisuan18181835245/my-model
tag	String	是	镜像tag，如 v1.0

响应示例：

```
{
  "code": 0,
  "data": {},
  "msg": "镜像删除成功"
}
```

4.1.9 提交容器产品为新镜像

接口地址：/agent/user/images/commitProduct

请求方式： POST

请求参数：

参数名	类型	必填	说明
productName	String	是	要提交的产品（容器）名称
newImageName	String	是	新镜像名（Harbor 仓库名，不含项目前缀与tag）
nodeIP	String	是	产品所在节点 IP

响应示例：

```
{
  "code": 0,
  "data": {},
  "msg": "提交成功"
}
```

4.1.10 获取容器规格

接口地址：/agent/user/products/spec

请求方式：GET（Query格式传参）

请求参数：

参数名	类型	必填	说明
gpuTypeId	String	否	GPU类型id，默认返回所有规格

响应示例:


```
{
  "code": 0,
  "data": [
    {
      "ip": "127.0.0.4", // 节点IP
      "integrationPort": 0,
      "zone": "测试一区", // 节点区域
      "priceId": 5, // 规格id
      "gpuType": "NVIDIA 4090", // GPU 型号
      "gpuTypeId": 8, // GPU 类型id
      "gpuTypeInfo": {
        "ID": 8,
        "CreatedAt": "2026-03-31T14:09:57.036+08:00",
        "UpdatedAt": "2026-03-31T14:09:57.036+08:00",
        "brand": "NVIDIA", // GPU 品牌
        "model": "4090", // GPU 类型
        "gpuMemory": 48, // GPU 显存
        "tFlopsValue": 330, // FP16
        "logo":
"uploads/file/9ceea81e5de699324754e8ac5b00d72d_20260331140940.png"
      },
      "inventory": {
        "ID": 5,
        "CreatedAt": "2026-03-31T16:26:49.125+08:00",
        "UpdatedAt": "2026-08-20T15:14:52.439+08:00",
        "node": "127.0.0.4",
        "gpuType": null,
        "gpuTypeId": 0,
        "available": 6, // 可用GPU数量
        "count": 8, // 节点总GPU数量
        "nodeType": null
      },
      "endTime": null,
      "price": 2.57, // 代理商折扣价格（/时）
      "arch": "x86", // 镜像架构（arm/x86）
      "cpuModle": "Intel@ Xeon@ Gold 6530", // CPU型号
      "cpuCount": 15, // CPU 核数
      "memory": 96, // 内存 （G）
      "localStorage": 500, // 数据盘（G）
      "sysStorage": 40, // 系统盘（G）
      "belong": 2,
      "belongAgent": []
    }
  ],
  "msg": "成功"
}
```

4.1.11 获取GPU类型

接口地址：/agent/user/products/type

请求方式：GET

请求参数：无

响应示例：


```
{
  "code": 0,
  "data": [
    {
      "ID": 10,
      "CreatedAt": "2026-03-31T14:15:22.194+08:00",
      "UpdatedAt": "2026-03-31T14:26:22.269+08:00",
      "brand": "NVIDIA", // 品牌
      "model": "1060", // 型号
      "gpuMemory": 3, // 显存
      "tFlopsValue": 111, // FP16
      "logo": "uploads/file/9ceea81e5de699324754e8ac5b00d72d_20260331142614.png"
    }
  ],
  "msg": "成功"
}
```


4.1.12 获取产品库存

接口地址：/agent/user/products/inventory

请求方式：GET（Query格式传参）

请求参数：

参数名	类型	必填	说明
ip	String	否	节点ip，默认返回所有节点库存信息

响应示例：


```
{
  "code": 0,
  "data": [
    {
      "ID": 2,
      "CreatedAt": "2026-03-31T14:35:19.763+08:00",
      "UpdatedAt": "2026-03-31T16:55:19.412+08:00",
      "node": "127.0.0.1", // 节点ip
      "gpuType": null,
      "gpuTypeId": 0,
      "available": 7, // 可用GPU数量
      "count": 8, // 节点GPU总量
      "nodeType": null
    }
  ],
```

```
  "msg": "成功"
}
```

4.2 产品相关接口

4.2.1 获取存储价格

接口地址：/agent/user/orders/getStoragePrice

请求方式：GET

请求参数：无

响应示例：

```
{
  "code": 0,
  "data": {
    "price": 2 //  代理商扩展存储价格（40G/小时）
  },
  "msg": "成功"
}
```

4.2.2 创建容器实例

接口地址：/agent/user/orders/create

请求方式：POST（Json格式传参）

请求参数：

参数名	类型	必填	说明
renewPolicy	String	是	续费策略（rolling/onetime）,包小时产品只能传rolling
durationHour	Integer	是	订购时长（单位：小时），1（包小时），24（包天） 720（包月）
node	String	是	节点ip
gpuCount	Integer	是	订购gpu数量
image	String	是	镜像完整名称（由镜像name:version组成）
ports	String	是	自定义容器端口，以","隔开(不开放80、443、8080端口)
spec	Integer	是	规格id
type	String	是	产品类型（容器：container）
expendStorage	Integer	否	扩展存储（仅支持（40/80/120）G/时）

请求参数示例：

```
{
  "renewPolicy": "rolling", // 续费策略 （rolling:自动续费，onetime:不自动续费）
  "durationHour": 1, // 订购时长（单位：小时，1（包小时），24（包天）720（包月））
  "node": "127.0.0.1", // 节点IP
  "gpuCount": 1, //订购gpu数量
  "image": "jiyun/cuda-jupyter-ssh:latest", // 镜像名称
  "ports": "22,8888,999,888,777,666", // 自定义端口
  "spec": 1, // 规格id
  "type": "container", // 类型
  "expendStorage": 0 // 扩展存储
}
```

响应示例：

```
{
  "code": 0,
  "data": {
    "ID": 1258, // 订单id
    "orderNo": "20260415155024649769"
  },
  "msg": "成功"
}
```

4.2.3 获取产品详情

接口地址：/agent/user/products/detail

请求方式：GET（Query格式传参）

请求参数：

参数名	类型	必填	说明
orderID	String	是	订单id
name	String	否	容器实例名称

响应示例：

```
{
  "code": 0,
  "data": {
    "product": {
      "ID": 2033,
      "CreatedAt": "2026-10-13T16:13:45.594+08:00",
      "UpdatedAt": "2026-10-13T16:13:47.341+08:00",
      "name": "8f30b16d18181835245",
      "node": "127.0.0.1",
      "orderID": 3011,
      "userName": "18181835245",
      "status": "running",
      "startTime": "2026-10-13T16:13:47.34+08:00",
      "endTime": "2026-10-13T17:13:47.34+08:00",
      "shutdownTime": null,
```

```
      "runingTime": null,
      "orderTime": 1,
      "singleDuration": 1,
      "singleDurationPrice": 540,
      "renewPolicy": "rolling",
      "type": "container",
      "gpuCount": 1,
      "cpuCount": 1,
      "memory": 16,
      "ports": "[{\"name\":\"SSH\",\"port\":\"22\"},
{\"name\":\"JupyterLab\",\"port\":\"8888\"},{\"name\":\"web
ui\",\"port\":\"7860\"}]",
      "command": null,
      "env": null,
      "image": "jiyun/cuda-jupyter-ssh:latest",
      "localStorage": 20,
      "sysStorage": 20,
      "expendStorage": 0,
      "info": {
        "container": {
          "config": {
            "AttachStderr": false,
            "AttachStdin": false,
            "AttachStdout": false,
            "Cmd": null,
            "Domainname": "",
            "Entrypoint": null,
            "Env": [
              "CONTAINER_VERSION=1",
              "JUPYTER_TOKEN=5b402648fd792584e1d212c05923bc2d",
              "MY_TOKEN=2ecfc8fc363201de"
            ],
            "ExposedPorts": {
              "22/tcp": {},
              "22/udp": {},
              "666/tcp": {},
              "666/udp": {},
              "777/tcp": {},
              "777/udp": {},
              "888/tcp": {},
              "888/udp": {},
              "8888/tcp": {},
              "8888/udp": {},
              "999/tcp": {},
              "999/udp": {}
            },
            "Hostname": "",
            "Image": "registry.jygpu.com/jiyun/cuda-jupyter-ssh:latest",
            "Labels": null,
            "OnBuild": null,
            "OpenStdin": true,
            "StdinOnce": false,
            "Tty": true,
            "User": "",
            "Volumes": null,
```

```
    "WorkingDir": ""
  },
  "containerName": "8f30b16d18181835245-1",
  "createTime": "2026-10-13 16:13:46",
  "hostConfig": {
    "AutoRemove": false,
    "Binds": ["/data/models:/model/:ro"],
    "BlkioDeviceReadBps": null,
    "BlkioDeviceReadIOps": null,
    "BlkioDeviceWriteBps": null,
    "BlkioDeviceWriteIOps": null,
    "BlkioWeight": 0,
    "BlkioWeightDevice": null,
    "CapAdd": null,
    "CapDrop": null,
    "Cgroup": "",
    "CgroupParent": "",
    "CgroupnsMode": "",
    "ConsoleSize": [0, 0],
    "ContainerIDFile": "",
    "CpuCount": 0,
    "CpuPercent": 0,
    "CpuPeriod": 0,
    "CpuQuota": 0,
    "CpuRealtimePeriod": 0,
    "CpuRealtimeRuntime": 0,
    "CpuShares": 0,
    "CpusetCpus": "1-1",
    "CpusetMems": "",
    "DeviceCgroupRules": null,
    "DeviceRequests": [
      {
        "Capabilities": [["gpu"]],
        "Count": 0,
        "DeviceIDs": ["GPU-f474d723-16a2-b5ab-273c-14ab589a9d04"],
        "Driver": "nvidia",
        "Options": null
      }
    ],
    "Devices": null,
    "Dns": null,
    "DnsOptions": null,
    "DnsSearch": null,
    "ExtraHosts": null,
    "GroupAdd": null,
    "IOMaximumBandwidth": 0,
    "IOMaximumIOps": 0,
    "IpcMode": "",
    "Isolation": "",
    "Links": null,
    "LogConfig": {
      "Config": null,
      "Type": ""
    },
    "MaskedPaths": null,
```

```
            "Memory": 17179869184,
            "MemoryReservation": 0,
            "MemorySwap": 0,
            "MemorySwappiness": null,
            "Mounts": [
              {
                "Source": "8f30b16d18181835245-1",
                "Target": "/data/",
                "Type": "volume"
              }
            ],
            "NanoCpus": 0,
            "NetworkMode": "",
            "OomKillDisable": null,
            "OomScoreAdj": 0,
            "PidMode": "",
            "PidsLimit": null,
            "PortBindings": {
              "22/tcp": [
                {
                  "HostIp": "",
                  "HostPort": "40255"
                }
              ],
              "22/udp": [
                {
                  "HostIp": "",
                  "HostPort": "40255"
                }
              ],
              "666/tcp": [
                {
                  "HostIp": "",
                  "HostPort": "40254"
                }
              ],
              "666/udp": [
                {
                  "HostIp": "",
                  "HostPort": "40254"
                }
              ],
              "777/tcp": [
                {
                  "HostIp": "",
                  "HostPort": "40258"
                }
              ],
              "777/udp": [
                {
                  "HostIp": "",
                  "HostPort": "40258"
                }
              ],
              "888/tcp": [
```

```
            {
              "HostIp": "",
              "HostPort": "40257"
            }
          ],
          "888/udp": [
            {
              "HostIp": "",
              "HostPort": "40257"
            }
          ],
          "8888/tcp": [
            {
              "HostIp": "",
              "HostPort": "40256"
            }
          ],
          "8888/udp": [
            {
              "HostIp": "",
              "HostPort": "40256"
            }
          ],
          "999/tcp": [
            {
              "HostIp": "",
              "HostPort": "40259"
            }
          ],
          "999/udp": [
            {
              "HostIp": "",
              "HostPort": "40259"
            }
          ]
        },
        "Privileged": false,
        "PublishAllPorts": false,
        "ReadonlyPaths": null,
        "ReadonlyRootfs": false,
        "RestartPolicy": {
          "MaximumRetryCount": 0,
          "Name": ""
        },
        "SecurityOpt": null,
        "ShmSize": 51539607552,
        "StorageOpt": {
          "size": "20GB"
        },
        "UTSMode": "",
        "Ulimits": null,
        "UsernsMode": "",
        "VolumeDriver": "",
        "VolumesFrom": null
      },
```

```
        "networkingConfig": {
          "EndpointsConfig": null
        },
        "platform": {
          "architecture": "",
          "os": ""
        },
        "version": 1
      }
    },
    "remark": null,
    "releaseTime": null,
    "sharedStorage": "",
    "statusDetails": null,
    "nodeInfo": {
      "ID": 57,
      "CreatedAt": "2026-08-12T18:59:41.356+08:00",
      "UpdatedAt": "2026-10-13T15:24:16.885+08:00",
      "name": "1060主机",
      "code": "local111",
      "ip": "127.0.0.1",
      "integrationPort": 2378,
      "priceId": 1,
      "zone": "1060专区",
      "gpuType": "NVIDIA 1060",
      "gpuTypeId": 10,
      "status": "Available",
      "endTime": null,
      "maintenanceDescription": null,
      "recoveryTime": null,
      "belong": 1,
      "tag": "共享节点1",
      "leaseTime": "2026-10-10T12:00:00+08:00",
      "terminateTime": "2026-11-12T12:00:00+08:00",
      "remark": "",
      "cardLockNum": null,
      "cardLockStartTime": null,
      "cardLockEndTime": null,
      "cardTag": {
        "tagBackgroundColor": "rgba(255, 6, 6, 1)",
        "tagBorderColor": "rgba(239, 0, 0, 1)",
        "tagText": "热力推荐",
        "tagTooltip": "算力闲忙优惠",
        "tagTextColor": "rgba(12, 12, 12, 1)"
      },
      "showCardTag": true,
      "belongAgent": []
    },
    "productTask": null
  },
  "pricing": {
    "ID": 1,
    "CreatedAt": "2026-03-31T14:28:57.255+08:00",
    "UpdatedAt": "2026-09-28T16:25:21.458+08:00",
    "gpuType": "NVIDIA 1060",
```

```
      "gpuTypeId": 10,
      "gpuModle": null,
      "cpuCount": 1,
      "cpuModle": "INTEL1060",
      "mem": 16,
      "localStorage": 20,
      "sysStorage": 20,
      "billingUnit": "时",
      "price": 6,
      "effectiveFrom": null,
      "effectiveTo": null,
      "gpuTypeInfo": {
        "ID": 0,
        "CreatedAt": "0001-01-01T00:00:00Z",
        "UpdatedAt": "0001-01-01T00:00:00Z",
        "brand": null,
        "model": null,
        "gpuMemory": null,
        "tFlopsValue": null,
        "logo": null
      },
      "isRecommend": true,
      "buttonDesc": "立即试用",
      "sort": 0,
      "discountVersion": "test",
      "arch": "x86"
    },
    "gpuTypeInfo": {
      "ID": 10,
      "CreatedAt": "2026-03-31T14:15:22.194+08:00",
      "UpdatedAt": "2026-03-31T14:26:22.269+08:00",
      "brand": "NVIDIA",
      "model": "1060",
      "gpuMemory": 3,
      "tFlopsValue": 111,
      "logo": "uploads/file/9ceea81e5de699324754e8ac5b00d72d_20260331142614.png"
    },
    "zone": "1060专区"
  },
  "msg": "成功"
}
```


4.2.4 获取产品列表

接口地址：/agent/user/products/list

请求方式：GET（Query格式传参）

请求参数：

参数名	类型	必填	说明
page	String	否	页码
pageSize	String	否	页量

参数名	类型	必填	说明
name	String	否	容器实例名称

响应示例:


```
{
  "code": 0,
  "data": {
    "list": [
      {
        "ID": 2033, // 产品ID
        "CreatedAt": "2026-10-13T16:13:45.594+08:00",
        "UpdatedAt": "2026-10-13T16:13:47.341+08:00",
        "name": "8f30b16d18181835245", // 容器实例名称
        "node": "222.212.85.181",
        "orderID": 3011, // 订单ID
        "userName": "18181835245",
        "status": "running",
        "startTime": "2026-10-13T16:13:47.34+08:00",
        "endTime": "2026-10-13T17:13:47.34+08:00",
        "shutdownTime": null,
        "runingTime": null,
        "orderTime": 1,
        "singleDuration": 1,
        "singleDurationPrice": 540,
        "renewPolicy": "rolling",
        "type": "container",
        "gpuCount": 1,
        "cpuCount": 1,
        "memory": 16,
        "ports": "[{\"name\":\"SSH\",\"port\":\"22\"},
{\"name\":\"JupyterLab\",\"port\":\"8888\"},{\"name\":\"web
ui\",\"port\":\"7860\"}]",
        "command": null,
        "env": null,
        "image": "jiyun/cuda-jupyter-ssh:latest",
        "localStorage": 20,
        "sysStorage": 20,
        "expendStorage": 0,
        "info": {
          "container": {
            "config": {
              "AttachStderr": false,
              "AttachStdin": false,
              "AttachStdout": false,
              "Cmd": null,
              "Domainname": "",
              "Entrypoint": null,
              "Env": [
                "CONTAINER_VERSION=1",
                "JUPYTER_TOKEN=5b402648fd792584e1d212c05923bc2d",
                "MY_TOKEN=2ecfc8fc363201de"
              ],
              "ExposedPorts": {
```

```
      "22/tcp": {},
      "22/udp": {},
      "666/tcp": {},
      "666/udp": {},
      "777/tcp": {},
      "777/udp": {},
      "888/tcp": {},
      "888/udp": {},
      "8888/tcp": {},
      "8888/udp": {},
      "999/tcp": {},
      "999/udp": {}
    },
    "Hostname": "",
    "Image": "registry.jygpu.com/jiyun/cuda-jupyter-ssh:latest",
    "Labels": null,
    "OnBuild": null,
    "OpenStdin": true,
    "StdinOnce": false,
    "Tty": true,
    "User": "",
    "Volumes": null,
    "WorkingDir": ""
  },
  "containerName": "8f30b16d18181835245-1",
  "createTime": "2026-10-13 16:13:46",
  "hostConfig": {
    "AutoRemove": false,
    "Binds": ["/data/models:/model/:ro"],
    "BlkioDeviceReadBps": null,
    "BlkioDeviceReadIOps": null,
    "BlkioDeviceWriteBps": null,
    "BlkioDeviceWriteIOps": null,
    "BlkioWeight": 0,
    "BlkioWeightDevice": null,
    "CapAdd": null,
    "CapDrop": null,
    "Cgroup": "",
    "CgroupParent": "",
    "CgroupnsMode": "",
    "ConsoleSize": [0, 0],
    "ContainerIDFile": "",
    "CpuCount": 0,
    "CpuPercent": 0,
    "CpuPeriod": 0,
    "CpuQuota": 0,
    "CpuRealtimePeriod": 0,
    "CpuRealtimeRuntime": 0,
    "CpuShares": 0,
    "CpusetCpus": "1-1",
    "CpusetMems": "",
    "DeviceCgroupRules": null,
    "DeviceRequests": [
      {
        "Capabilities": [["gpu"]],
```

```
            "Count": 0,
            "DeviceIDs": ["GPU-f474d723-16a2-b5ab-273c-14ab589a9d04"],
            "Driver": "nvidia",
            "Options": null
          }
        ],
        "Devices": null,
        "Dns": null,
        "DnsOptions": null,
        "DnsSearch": null,
        "ExtraHosts": null,
        "GroupAdd": null,
        "IOMaximumBandwidth": 0,
        "IOMaximumIOps": 0,
        "IpcMode": "",
        "Isolation": "",
        "Links": null,
        "LogConfig": {
          "Config": null,
          "Type": ""
        },
        "MaskedPaths": null,
        "Memory": 17179869184,
        "MemoryReservation": 0,
        "MemorySwap": 0,
        "MemorySwappiness": null,
        "Mounts": [
          {
            "Source": "8f30b16d18181835245-1",
            "Target": "/data/",
            "Type": "volume"
          }
        ],
        "NanoCpus": 0,
        "NetworkMode": "",
        "OomKillDisable": null,
        "OomScoreAdj": 0,
        "PidMode": "",
        "PidsLimit": null,
        "PortBindings": {
          "22/tcp": [
            {
              "HostIp": "",
              "HostPort": "40255"
            }
          ],
          "22/udp": [
            {
              "HostIp": "",
              "HostPort": "40255"
            }
          ],
          "666/tcp": [
            {
              "HostIp": "",
```

```
        "HostPort": "40254"
      }
    ],
    "666/udp": [
      {
        "HostIp": "",
        "HostPort": "40254"
      }
    ],
    "777/tcp": [
      {
        "HostIp": "",
        "HostPort": "40258"
      }
    ],
    "777/udp": [
      {
        "HostIp": "",
        "HostPort": "40258"
      }
    ],
    "888/tcp": [
      {
        "HostIp": "",
        "HostPort": "40257"
      }
    ],
    "888/udp": [
      {
        "HostIp": "",
        "HostPort": "40257"
      }
    ],
    "8888/tcp": [
      {
        "HostIp": "",
        "HostPort": "40256"
      }
    ],
    "8888/udp": [
      {
        "HostIp": "",
        "HostPort": "40256"
      }
    ],
    "999/tcp": [
      {
        "HostIp": "",
        "HostPort": "40259"
      }
    ],
    "999/udp": [
      {
        "HostIp": "",
        "HostPort": "40259"
```

```
                }
              ]
            },
            "Privileged": false,
            "PublishAllPorts": false,
            "ReadonlyPaths": null,
            "ReadonlyRootfs": false,
            "RestartPolicy": {
              "MaximumRetryCount": 0,
              "Name": ""
            },
            "SecurityOpt": null,
            "ShmSize": 51539607552,
            "StorageOpt": {
              "size": "20GB"
            },
            "UTSMode": "",
            "Ulimits": null,
            "UsernsMode": "",
            "VolumeDriver": "",
            "VolumesFrom": null
          },
          "networkingConfig": {
            "EndpointsConfig": null
          },
          "platform": {
            "architecture": "",
            "os": ""
          },
          "version": 1
        }
      },
      "remark": null,
      "releaseTime": null,
      "sharedStorage": "",
      "statusDetails": null,
      "nodeInfo": null,
      "productTask": null
    }
  ],
  "total": 1,
  "page": 1,
  "pageSize": 10
},
"msg": "获取成功"
}
```


4.2.5 容器开机

接口地址：/agent/user/products/start

请求方式：PATCH（Query格式传参）

请求参数：

参数名	类型	必填	说明
name	String	是	容器实例名称

响应示例:

```
{
  "code": 0,
  "data": {},
  "msg": "success"
}
```

4.2.6 容器关机

接口地址：/agent/user/products/shutdown

请求方式：PATCH（Query格式传参）

请求参数:

参数名	类型	必填	说明
name	String	是	容器实例名称

响应示例:

```
{
  "code": 0,
  "data": {},
  "msg": "success"
}
```

4.2.7 更新续费策略（包天/包月产品调用）

接口地址：/agent/user/products/updateRenewPolicy

请求方式：PUT（Json格式传参）

请求参数:

参数名	类型	必填	说明
id	Integer	是	产品id
renewPolicy	String	是	续费策略，rolling（自动续费）/onetime（到期关机）

响应示例:

```
{
  "code": 0,
  "data": {},
  "msg": "更新成功"
}
```

4.2.8 更换镜像

接口地址：/agent/user/products/updateImage

请求方式：PUT（Json格式传参）

请求参数：

参数名	类型	必填	说明
productName	String	是	容器实例名称
image	String	是	镜像完整名称（由镜像name:version组成）
ports	String	是	自定义容器端口，以","隔开(不支持80、443、8080端口)

响应示例：

```
{
  "code": 0,
  "data": {},
  "msg": "更换产品镜像成功"
}
```

4.2.9 容器续费 （包天/包月产品主动续费时调用）

接口地址：/agent/user/products/renew

请求方式：PUT（Json格式传参）

请求参数：

参数名	类型	必填	说明
orderID	Integer	是	订单id
renewTime	Integer	是	续费时长（数值为天数*24，1-29天，单位：小时）

响应示例：

```
{
  "code": 0,
  "data": {},
  "msg": "续费成功"
}
```

4.2.10 释放实例

温馨提示：容器释放会删除容器和数据，且该操作不可逆！

接口地址：/agent/user/orders/complete

请求方式：POST（Json格式传参）

请求参数：

参数名	类型	必填	说明
orderID	Integer	是	订单id

响应示例：

```
{
  "code": 0,
  "data": {},
  "msg": "完成订单成功"
}
```

4.3 代理商消费情况

4.3.1 获取代理商消费详情

接口地址：/agent/user/orders/getTransactionStatistics

请求方式：GET

请求参数：无

响应示例：

```
{
  "code": 0,
  "data": {
    "balance": 19796, // 代理商账户当前余额  （单位：分）
    "spent": 7194, // 代理商累计消费额     （单位：分）
    "topUpAmount": 12786465, // 代理商账户总充值金额（单位：分）
    "discount": 0.7, // 代理商当前折扣率
    "nextDiscountAmount": 2992806, // 下一梯度所需消费额  （单位：分）
    "nextDiscount": 0.6 // 下一梯度折扣率
  },
  "msg": "成功"
}
```

5. 产品说明

5.1.1 产品状态

计费类型	购买周期内	到期后
按时计费	开机成功后为running，关机后为paused，会归还gpu资源	到期后为paused，归还gpu资源
包天/包月	开机成功后为running，关机后为sleep，不归还gpu资源	到期后为paused，归还gpu资源

5.1.2 产品计费

续费策略：rolling-到期自动续费，onetime-到期自动停止

计费类型	周期	描述
按时计费	1小时	创建时为rolling，关机状态则释放资源，不续费；开机状态下自动续费
包天	1-29天	rolling时到期按1天进行自动续费，onetime时到期自动停止，支持主动续费，主动续费时间为：天数*24小时
包月	1-11月	rolling时到期按1月进行自动续费，onetime时到期自动停止，支持主动续费，主动续费时间为：天数*24小时

5.1.3 架构

架构：Nvidia机器和华为昇腾机器在架构方面有所不同，因此机器使用的镜像也有所差异

机器	镜像
华为910B	创建实例或更换镜像时，仅可使用镜像列表中架构（arch）为 arm 的镜像
NVIDIA型号	创建实例或更换镜像时，仅可使用镜像列表中架构（arch）为 x86 的镜像

5.1.4 接口使用

序号	操作	描述
1	获取镜像、容器规格、库存信息类型等信息	调用资源接口获取镜像、规格、库存、成本等信息
2	创建容器实例	调用创建接口，传入所需参数创建容器，获取返回的订单id
3	获取产品连接信息	根据订单id调用产品详情接口，提取相关连接信息
4	操作实例	调用开机、关机、更新续费策略、更换镜像等接口
5	产品续费	rolling状态的产品到期自动续费，主动续费接口用于包天/包月产品进行调用
5	释放产品	容器实例关机后默认保留10天，如需提前释放可调用此接口，会删除容器和数据

5.1.5 镜像仓库相关说明

镜像仓库开通：代理商在JYGPU平台上开通或调用开通镜像仓库的接口，即可开通镜像仓库，其下用户才可使用镜像仓库和保存容器镜像的功能。

镜像仓库使用：代理商的所有用户在保存镜像时需要接口严格拼接镜像名称，才能使其用户正常使用。如代理商的JYGPU仓库项目为zhisuan1800011111，其有用户手机号为16666666666，则用户保存镜像时间需要传参为zhisuan1800011111/16666666666-镜像名称:tag,保存成功后其用户16666666666才可见该镜像。用户在创建容器时，选择镜像的时候调用获取代理商镜像列表的接口，获取该代理商镜像列表，实际用户使用的时候需要根据用户实际手机号对镜像列表进行二次筛选，确保镜像的隔离性。

连接	描述
SSH	获取产品详情中节点ip、PortBindings下22端口的映射端口，密码为MY_TOKEN字段的值（示例：ssh root@127.0.0.1 -p 45257 ）
JUPYTER	获取产品详情中节点ip、PortBindings下8888端口的映射端口，密码为JUPYTER_TOKEN字段的值

5.1.56连接信息

获取产品详情：产品创建成功后可根据订单id查询产品详情，获取连接信息

连接	描述
SSH	获取产品详情中节点ip、PortBindings下22端口的映射端口，密码为MY_TOKEN字段的值（示例：ssh root@127.0.0.1 -p 45257 ）
JUPYTER	获取产品详情中节点ip、PortBindings下8888端口的映射端口，密码为JUPYTER_TOKEN字段的值

6. 使用建议

6.1 调用频率

- 常规接口：建议每分钟不超过60次
- 监控接口：建议每分钟不超过30次
- 批量操作：建议使用批量接口，减少请求次数

6.2 错误处理

- 遇到401错误：重新获取Token
- 遇到429错误：降低调用频率
- 遇到500错误：记录错误信息并重试

6.3 安全建议

- 不要在代码中硬编码API密钥
- 使用HTTPS协议进行通信
- 定期轮换API密钥
- 限制API访问IP

7. 联系我们

- **官方网站**：https://www.jygpu.com
- **文档中心**：https://doc.jygpu.com
- **客服热线**：400-028-0032