



链滴

矩池云如何自定义端口，访问自己的 web 项目

作者: [matpool](#)

原文链接: <https://ld246.com/article/1668567843307>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

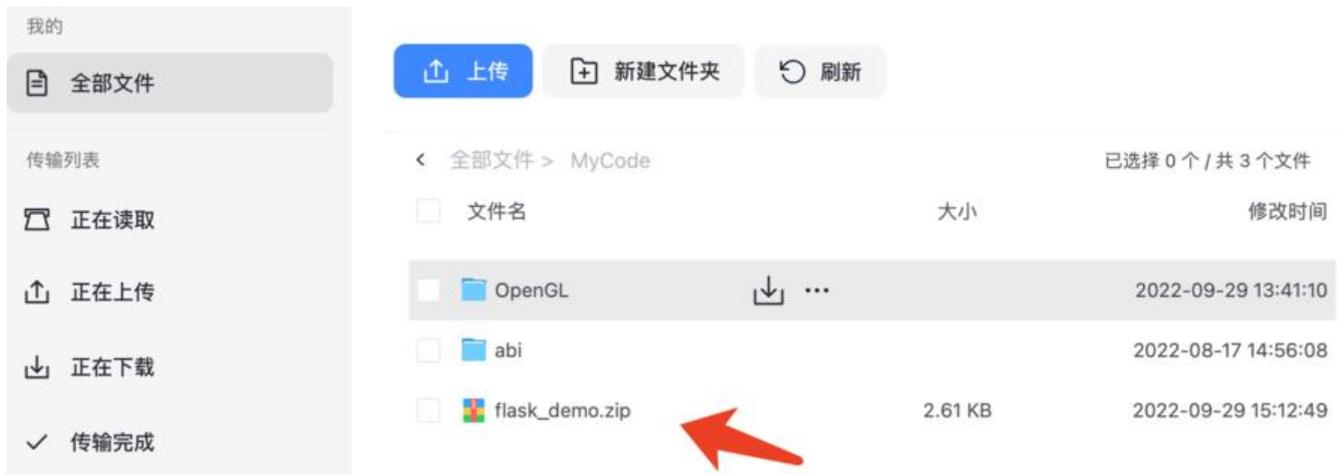
本文将给您介绍如何在矩池云租用服务器的时候自定义端口，并将您的 web 项目部署到自定义端口最后实现在本地通过自定义端口对应链接访问服务。

上传代码和数据

首先，您需要将本地的项目代码和数据上传到矩池云网盘。这里为了方便您测试使用，测试案例 flask demo 文件大家可以[点击这里下载](#)。

下载后，可以直接通过矩池云网盘上传，目前矩池云网盘客户端支持文件/文件夹上传，具体安装使用方法[点击这里查看](#)。

上传完成后，点下刷新按钮，即可看到您上传的测试 demo 了。



租用机器

选择一个您需要的环境（这里我选择的 JAX 这个系统镜像），然后点击 **高级功能**，在自定义端口输入您想自定义的端口，比如下图中自定义端口为：5000，另外默认的两个端口分别是 ssh 链接的（22 口）和 jupyterlab 链接的（8888 端口）。

JAX x

JAX
预装: Ubuntu20.04, Python 3.9, CUDA 11.1, cuDNN 8, NVCC, JAX 0.3.17, VNC

预设工具

关 开启VNC

设置公钥

ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQD:

自定义设置

自定义端口

SSH	22	镜像默认
HTTP	8888	镜像默认
HTTP	5000	<input type="button" value="-"/>



环境变量

定义环境变量，分号分隔，如: FOO=bar;BAZ=tic

GPU数 1

*约可租借 <629 小时

下单: ¥1.00/小时

启动访问服务

租用机器启动成功后，会有您自定义端口的访问链接，由于您服务还没有运行部署到对应端口，所以接是不可访问的。

运行中

常用信息

* 以下链接非长期有效，偶尔会因网络问题而变更

<	HTTP 5000端口	SSH	JupyterLab	PyCharm	VSCoc	>
提示	访问此端口链接前，请确保已经启动相应的服务					
访问链接 (URL)	https://hz-t		?token=29V9mJxluK			
域名 (Hostname)	hz-t3.matpool.com					
端口 (Port)	3					

解压文件

选择您习惯的远程连接方式进入机器，这里使用 [JupyterLab](#) 作为示例（使用起来最简单），点击 [JupyterLab](#) 中的 [点击进入](#)，即可进入 [JupyterLab](#) 页面，我们新建一个 [Terminal](#)，然后进入项目所在目录输入解压指令。（这里默认您的测试 demo 存放在 /mnt/MyCode/ 下，您也可以改成其他的）

```
cd /mnt/MyCode/  
ls  
unzip flask_demo.zip
```

```
(myconda) root@ZB1A5J:/# cd /mnt/MyCode/  
(myconda) root@ZB1A5J:/mnt/MyCode# ls  
OpenGL abi flask_demo.zip  
(myconda) root@ZB1A5J:/mnt/MyCode# unzip flask_demo.zip  
Archive: flask_demo.zip  
  creating: flask_demo/  
    creating: flask_demo/img/  
  inflating: flask_demo/index.html  
  inflating: flask_demo/flask_demo.py  
(myconda) root@ZB1A5J:/mnt/MyCode# █
```

配置环境

测试 demo 很简单，只需要额外安装 flask 包即可，输入 pip 指令即可安装，如果出现截图中安装的时候找不到包的情况，可以换个 pip 源试试。

```
pip install flask  
# 切换 pip 源  
bash /public/script/switch_pip_source.sh
```

```
(myconda) root@ZB1A5J:/mnt/MyCode# pip install flask  
Looking in indexes: https://mirrors.sjtug.sjtu.edu.cn/pypi/web/simple  
WARNING: Skipping page https://mirrors.sjtug.sjtu.edu.cn/pypi/web/simple/flask/ because the GET request got Content-Type: application/octet-stream.The only supported Content-Type is text/html  
ERROR: Could not find a version that satisfies the requirement flask (from versions: none)  
ERROR: No matching distribution found for flask  
WARNING: Skipping page https://mirrors.sjtug.sjtu.edu.cn/pypi/web/simple/pip/ because the GET request got Content-Type: application/octet-stream.The only supported Content-Type is text/html  
(myconda) root@ZB1A5J:/mnt/MyCode# bash /public/script/switch_pip_source.sh  
请选择您要切换的源的数字编号，然后按回车  
(0) 阿里云(aliyun)  
(1) 中国科技大学(ustc)  
(2) 豆瓣(douban)  
(3) 清华大学(tsinghua)  
(4) 腾讯云(tencent)  
(5) 浙江大学(zju)  
(6) 网易(163)  
(7) 华为云(huawei)  
(8) 北京外国语大学(bfsu)  
(9) 上海交通大学(sjtug)  
(10) 南京大学(nju)  
3  
切换完成  
(myconda) root@ZB1A5J:/mnt/MyCode# pip install flask  
Looking in indexes: https://pypi.tuna.tsinghua.edu.cn/simple/  
Collecting flask  
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/0f/43/15f4f9ab225b0b25352412e8daa3d0e3d135fcf5e127070c74c3632e8b4c/flask-2.2.2-py3-none-any.whl (101 kB)  
-----  
Collecting itsdangerous>=2.0  
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/68/5f/447e04e28f47465eeab35bd408b7ebaaae207f48b7136c5a7267a30ae/itsdangerous-2.1.2-py3-none-any.whl (15 kB)  
Requirement already satisfied: importlib-metadata>=3.6.0 in /root/miniconda3/envs/myconda/lib/python3.9/site-packages (from flask) (4.12.0)  
Collecting Jinja2>=3.0  
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/bc/c3/f068337a370801f372f2f8f6bad74a5c140f6fda3d9de154052708dd3c65/Jinja2-3.1.2-py3-none-any.whl (133 kB)  
-----  
Collecting click>=8.0  
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/c2/f1/d59e28c642d583f7dacf5b1e0965d0e0b218e0186d7858ac5233dce840/click-8.1.3-py3-none-any.whl (96 kB)  
-----  
Requirement already satisfied: Werkzeug>=2.2.2 in /root/miniconda3/envs/myconda/lib/python3.9/site-packages (from flask) (2.2.2)  
Requirement already satisfied: zipp>=0.5 in /root/miniconda3/envs/myconda/lib/python3.9/site-packages (from importlib-metadata>=3.6.0->flask) (3.8.1)  
Requirement already satisfied: MarkupSafe>=2.0 in /root/miniconda3/envs/myconda/lib/python3.9/site-packages (from Jinja2>=3.0->flask) (2.1.1)  
Installing collected packages: Jinja2, itsdangerous, click, flask  
Successfully installed Jinja2-3.1.2 click-8.1.3 flask-2.2.2 itsdangerous-2.1.2  
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
```

启动访问项目

Jupyterlab 页面新建一个 Terminal，输入 cd 指令先进入项目目录，然后输入指令运行程序，我们可以看到程序已经运行在了 5000 端口上。

注意：如果想要通过外网能访问到服务器启动的服务，代码里设置的 web host 必须为：0.0.0.0。

```
cd /mnt/MyCode/flask_demo
```

python flask_demo.py

```
(myconda) root@. /mnt/MyCode/flask_demo# python flask_demo.py
* Serving Flask app 'flask_demo'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://172.17.0.4:5000
Press CTRL+C to quit
```

启动成功后，我们点击租用界面 5000 端口对应的访问链接即可访问我们的服务了。



如上图所示，您访问您自己自定义端口链接时，可以不带 token 参数。