

Guia para rodar o Jupyter Notebook no Splinter e no Cosmo0

Jordany Vieira e Alessandro Marins
22 de dezembro de 2023

Introdução

Nosso objetivo neste texto é construir um guia prático para a rodar o jupyter notebook em um servidor externo e manipulá-lo no browser do computador pessoal. Aqui, está feito para o cluster SPLINTER da University College London (UCL) e para o Cosmo0 do grupo do BINGO na USP. A maior diferença entre o SPLINTER e o Cosmo0, no quesito acesso, é que no SPLINTER são necessários dois acessos. Primeiro ao servidor e depois ao cluster propriamente dito. Chamaremos cada acesso da máquina pessoal ao local externo por "túnel". Assim, enquanto no Cosmo0 será necessário um único túnel, no SPLINTER serão necessários dois túneis para rodar o jupyter.

Por todo esse guia será assumido que você usa alguma distribuição Linux.

SPLINTER

Orientações Preliminares

Primeiramente, é preciso acessar a sua conta no SPLINTER. Assumirei aqui que o seu usuário seja *user_X*. Assim, primeiro:

» `ssh user_X@zuserver2.star.ucl.ac.uk`

Digite a senha de acesso ao servidor. Depois, acesse o cluster:

» `ssh user_X@splinter-login.star.ucl.ac.uk`

Digite a senha de acesso ao cluster.

Agora é necessário criar um arquivo:

» touch jupyter-slurm.sh

E editá-lo.

» vim jupyter-slurm.sh

Cuidado, nesse momento você acessará o arquivo (ainda vazio) pelo ambiente **vim**. Sempre que quiser editar um arquivo vim é preciso digitar a letra *i*¹. Após editar o arquivo digite *Esc* para sair do modo edição e *:wq!*, para forçar (!) o ambiente (sobre)escrever a edição (w) e fechar o ambiente vim (q). Aqui, você copiará o seguinte trecho:

```
#!/bin/bash
#SBATCH - -partition COMPUTE
#SBATCH - -nodes 1
#SBATCH - -ntasks-per-node 1
#SBATCH - -mem-per-cpu 2G
#SBATCH - -time 1-0:00:00
#SBATCH - -job-name jupyter-notebook
#SBATCH - -output jupyter-notebook-%J.log
#Based on https://docs.ycrc.yale.edu/clusters-at-yale/guides/jupyter/ XDG_RUNTIME_DIR="
port=$(shuf -i8000-9999 -n1)
node=$(hostname -s)
user=$(whoami)
cluster=$(hostname -f | awk -F"." '{print $2}')

# print tunneling instructions jupyter-log
echo -e "To connect:
ssh -N -L $port:$node:$port $user@splinter-login.star.ucl.ac.uk

Use a Browser on your local machine to go to:
localhost:$port (prefix w/ https:// if using password)

Remember to scancel job when done. Check output below for access token if you
need it. "

eval "$(/share/apps/anaconda/3-2019.07/bin/conda shell.bash hook)"
conda activate /share/data1/amarins/env/2
srun -n1 jupyter-notebook - -no-browser - -port=$port --ip=$node
```

Após criar o arquivo jupyter-slurm.sh, precisamos que você abra um novo terminal (*Crtl+Alt+T*) e que você esteja na home do seu computador pessoal para criar um novo arquivo.

¹Em terminais linux, comumente, cópias de linhas são feitas pelo comando *Shift+Ctrl+V*; e colas, *Shift+Ctrl+C*.

²Aqui, você deve colocar a localização do seu ambiente Anaconda.

» touch config

e o acesse

» vim config

Nele você terá de copiar o seguinte trecho (lembrando que estamos assumindo que o seu usuário seja user_X):

```
Host splinterjump
  User user_X
  HostName splinter-login.star.ucl.ac.uk
  ProxyCommand ssh user_X@zuserver1.star.ucl.ac.uk -W %h:%p
```

A identificação é composta por quatro espaçamentos.

Agora, será necessário (caso não haja) um ambiente Anaconda (conda) para trabalhar na sua conta no SPLINTER. Digamos que criaremos uma pasta na sua home no SPLINTER para deixar o conda a ser ativado. Sua home é: "/home/user_X". Crie, p.ex., o diretório "env", para colocar o ambiente conda:

» mkdir env

Agora crie o ambiente

» conda create - -prefix /home/user_X/env python=2.7.17 pip

Sempre que for usar um código qualquer que precise de pacotes que você tem e instalou no seu ambiente conda é preciso ativá-lo. Se você quiser instalar pacotes nele, via pip, também basta estar ativo no ambiente, como verevemos na próxima seção.

Rodar um jupyter no SPLINTER

Agora, o que é preciso é deixar um jupyter-notebook ativo na fila do cluster para que ele possa ser acessado e executado remotamente. Para isso, sempre que for ativá-lo, primeiro carregue os seguintes módulos:

» module load mpi/openmpi-x86_64

» module load dev_tools/oct2018/python-Anaconda-3-5.3.0

que ativará o ambiente base com alguns pacotes necessários na sua área de trabalho. Em seguida, ativamos o conda

» conda activate /home/user_X/env/

Agora é preciso colocar um jupyter-notebook na fila do cluster. Para isso, basta digitar:

» sbatch jupyter-slurm.sh

Em seguida, para verificar que está na fila, digite

» `qstat`

e verifique que ele está ativo (R). Se sim, será criado um arquivo de extensão ".log" na sua home no SPLINTER. Digamos que seja: `jupyter-notebook-XXXXXX.log`. Abra-o no terminal:

» `more jupyter-notebook-XXXXXX.log`

Esse arquivo contém informações necessárias para os próximos passos. Ele estará parecido com (não exatamente igual à):

To connect:

`ssh -N -L 9585:compute-0-29:9585 user_X@splinter-login.star.ucl.ac.uk`

Use a Browser on your local machine to go to:

`localhost:9585` (prefix w/ `https://` if using password)

Remember to scancel job when done. Check output below for access token if you need it.

[I 17:03:02.712 NotebookApp] JupyterLab extension loaded from `/share/apps/anaconda/3-2019.03/lib/python3.7/site-packages/jupyterlab`

[I 17:03:02.712 NotebookApp] JupyterLab application directory is `/share/apps/anaconda/3-2019.03/share/jupyter/lab`

[I 17:03:02.714 NotebookApp] Serving notebooks from local directory: `/home/user_X`

[I 17:03:02.714 NotebookApp] The Jupyter Notebook is running at:

[I 17:03:02.714 NotebookApp] `http://compute-0-29:9585/?token=3e49997daa2daa7d6908bb09ecdfa2d1c08c58bb0995ce67`

[I 17:03:02.714 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).

[C 17:03:02.721 NotebookApp]

To access the notebook, open this file in a browser:

`file:///home/user_X/.local/share/jupyter/runtime/nbserver-14559-open.html`

Or copy and paste one of these URLs:

`http://compute-0-29:9585/?token=3e49997daa2daa7d6908bb09ecdfa2d1c08c58bb0995ce67`

O valor **9585**, na segunda linha, é a porta de acesso que foi liberada aqui para o jupyter (ela varia de valor). Copie esse valor, por generalidade chamarei 9585 de XXXX, representando que pode ser outro valor. Copie essa segunda linha até o segundo valor da porta, no caso, digamos que seja: `ssh -N -L XXXX:compute-0-YY:XXXX`. **Vide que acima, YY é 29, mas pode variar.**

Deixe esse terminal, com o arquivo aberto, e acesse o terminal com acesso ao seu computador pessoal. Digite:

» `ssh -N -L XXXX:compute-0-YY:XXXX splinterjump`

Com o seu valor correto da porta que lhe foi fornecida. Será solicitado a você duas senhas a do servidor (primeiro túnel) e para o cluster (segundo túnel). Após digitá-los não aparecerá mais nada no terminal, ficará somente a palavra de solicitação de senhas (não feche esse terminal senão a conexão será cortada). A conexão entre o computador pessoal e o cluster foi feita, agora é preciso abrir um browser e usar a porta no localhost. Abra um browser e digite: *localhost:XXXX*. Abrirá uma tela de login e será solicitado um *token*, que está no arquivo de extensão ".log", provavelmente na última linha. Basta copiar o token (somente o o valor do token, ou seja, tudo que vem depois do "token=") e colar no espaço em branco no browser.

Pronto, ele possibilitará, enfim, que você use o jupyter (rodando no cluster) remotamente. Você poderá acessar códigos com extensão ".ipynb" do SPLINTER ou criar códigos. Não conseguirá acessar códigos do seu computador pessoal. Caso isso seja necessário, será preciso enviar o código ao SPLINTER. Para isso, pode-se usar o comando *scp*.

Para cancelar um "job" no SPLINTER é necessário digitar *qstat* e ver o número à esquerda na tabela (primeira coluna), que digamos que seja YYYYYY, e digitar *qdel YYYYYY*.

Cosmo0

Orientações Preliminares

O Cosmo0 é um servidor do grupo BINGO instalado na Universidade de São Paulo, na cidade de São Paulo. Nele poderemos usar tanto jupyter-notebook quanto o jupyterLab de maneira remota, como veremos.

Primeiramente, supondo que você já possua uma conta no servidor e que o seu usuário é (novamente) **user_X**. Então, entre na sua conta:

```
» ssh user_X@cosmo0.if.usp.br
```

e digite sua senha de acesso ao servidor. Em seguida, precisamos construir nosso ambiente de trabalho, que pode ser conda ou não, a cargo de cada um. Criaremos o ambiente na sua home (/home/user_X). Digamos que chamaremos o ambiente de **env**:

```
» python3 -m venv env
```

Você poderá olhar para o que nele será feito usando o comando *ls* no terminal. Sempre que for necessário usá-lo precisamos ativá-lo, e para isso:

```
» source /home/user_X/env/bin/activate
```

Você estará no ambiente. Qualquer pacote python3 necessário para o seu trabalho poderá ser instalado pelo pip e estará exclusivamente no seu ambiente. Por ora, isso é a parte preliminar. Digamos que você vá instalar somente os pacotes básicos mais os pacotes do jupyter-notebook e do jupyterLab (esse é opcional). Então, no seu ambiente,

```
(env)» python3 -m pip install setuptools cython wheel pillow
```

```
(env)» python3 -m pip install --upgrade setuptools
```

```
(env)» python3 -m pip install numpy scipy matplotlib healpy astropy pandas
```

(env)» `python3 -m pip install ipykernel jupyter jupyterlab-server jupyterlab`

Agora, vejamos como fazer um túnel entre o computador pessoal e o cosmo0 para rodar jupyter remontadamente usando o cosmo0.

Rodar um jupyter no Cosmo0

O tunelamento entre o computador pessoal e o servidor é simples. precisamos estabelecer qual porta do computador pessoal e qual do servidor usaremos (cuidado, aqui, verifique se a porta do servidor não está sendo utilizada. Caso sim, use outro valor). Valor padrão dado às portas é 8888, aqui eu as chamarei de XXXX (porta do computador pessoal) e YYYY (porta no servidor). Assim, entraremos no servidor da seguinte maneira:

» `ssh -L XXXX:localhost:YYYY user_X@cosmo0.if.usp.br`

Você entrará no Cosmo0. Agora, ative o ambiente,

» `source /home/user_X/env/bin/activate`

e rode o jupyter, ou o jupyterLab, forçando a ele assumir a porta YYYY

» `jupyter notebook -no-browser -port YYYY`

ou

» `jupyter lab -no-browser -port YYYY`

Aparecerá algo como³

```
[I 2021-03-25 08:31:43.413 ServerApp] jupyterlab | extension was successfully linked.
[I 2021-03-25 08:31:43.597 ServerApp] nbclassic | extension was successfully linked.
[I 2021-03-25 08:31:43.638 LabApp] JupyterLab extension loaded from /home/user_X/env/lib/python3.6/site-packages/jupyterlab
[I 2021-03-25 08:31:43.638 LabApp] JupyterLab application directory is /home/user_X/env/share/jupyter/lab
[I 2021-03-25 08:31:43.640 ServerApp] jupyterlab | extension was successfully loaded.
[I 2021-03-25 08:31:43.643 ServerApp] nbclassic | extension was successfully loaded.
[I 2021-03-25 08:31:43.643 ServerApp] Serving notebooks from local directory: /home/user_X
[I 2021-03-25 08:31:43.643 ServerApp] Jupyter Server 1.2.2 is running at:
[I 2021-03-25 08:31:43.643 ServerApp] http://localhost:YYYY/lab?token=ae265c70aaf1d742560d0f2e390e1ec8a3a661
[I 2021-03-25 08:31:43.643 ServerApp] or http://127.0.0.1:YYYY/lab?token=ae265c70aaf1d742560d0f2e390e1ec8a3a66
[I 2021-03-25 08:31:43.643 ServerApp] Use Control-C to stop this server and shut down
all kernels (twice to skip confirmation).
[C 2021-03-25 08:31:43.647 ServerApp]
```

To access the server, open this file in a browser:

`file:///home/user_X/.local/share/jupyter/runtime/jpserver-10637-open.html`

Or copy and paste one of these URLs:

³O exemplo é para o jupyterlab. Contudo, é igual para o jupyter-notebook

`http://localhost:YYYY/lab?token=ae265c70aaf1d742560d0f2e390e1ec8a3a661788db475e4`
or `http://127.0.0.1:YYYY/lab?token=ae265c70aaf1d742560d0f2e390e1ec8a3a661788db475e4`

copie a parte **`http://localhost:YYYY/lab?token=ae265c70aaf1d742560d0f2e390e1ec8a3a661788db475e4`**. Em seguida, abra um browser de sua preferência e cole nele essa parte. Você será direcionado ao jupyter(lab) no seu browser conectado ao Cosmo0. Agora é só usar.

Referências

- [1] <https://www.digitalocean.com/community/tutorials/how-to-install-python-3-and-set-up-a-programming-environment-on-an-ubuntu-18-04-server>
- [2] <https://www.digitalocean.com/community/tutorials/how-to-install-run-connect-to-jupyter-notebook-on-remote-server-pt>
- [3] <https://liyanxu.blog/2018/12/08/running-jupyter-notebook-on-remote-server/>