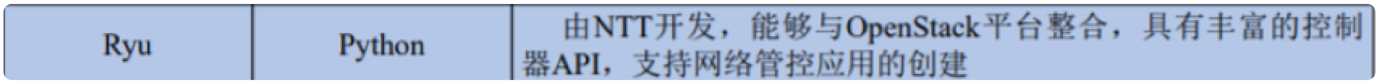
# RYU的安装

# 一：Ryu是主流SDN开源控制器之一



| | | |
|---|---|---|
| Ryu | Python | 由NTT开发，能够与OpenStack平台整合，具有丰富的控制器API，支持网络管控应用的创建 |

## （一）推文（摘录自）

https://ryu.readthedocs.io/en/latest/

https://www.sdnlab.com/1785.html

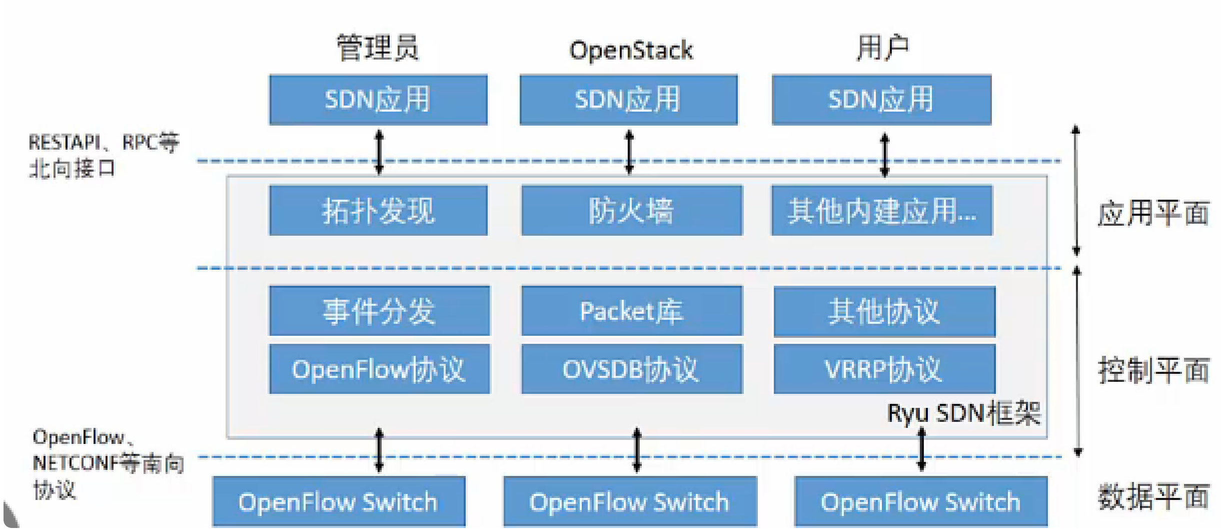
## （二）Ryu是什么

```
Ryu is a component-based software defined networking framework.    Ryu是一个基于组件的软件定义网络框架

Ryu provides software components with well defined API's that make it easy for developers to create new network management and control applications. Ryu为软件组件提供了定义良好的API，使开发人员能够轻松地创建新的网络管理和控制应用程序。
Ryu supports various protocols for managing network devices, such as OpenFlow, Netconf, OF-config, etc.    支持管理网络设备的多种协议
About OpenFlow, Ryu supports fully 1.0, 1.2, 1.3, 1.4, 1.5 and Nicira Extensions.    所支持的OpenFLow协议版本

All of the code is freely available under the Apache 2.0 license. Ryu is fully written in Python.    所有代码在Apache2.0许可下都是免费的，Ryu是使用Python开发的
```

## （三）Ryu的架构



# 二：Ryu的安装

## （一）git远程获取文件

Shell 复制代码

```
git clone git://github.com/osrg/ryu.git
```

或者

Shell 复制代码

```
git clone https://github.com/faucetsdn/ryu.git
```

## （二）安装Ryu依赖环境

Shell 复制代码

```
cd ryu    需要先进入文件夹中
sudo pip install -r tools/pip-requires
```

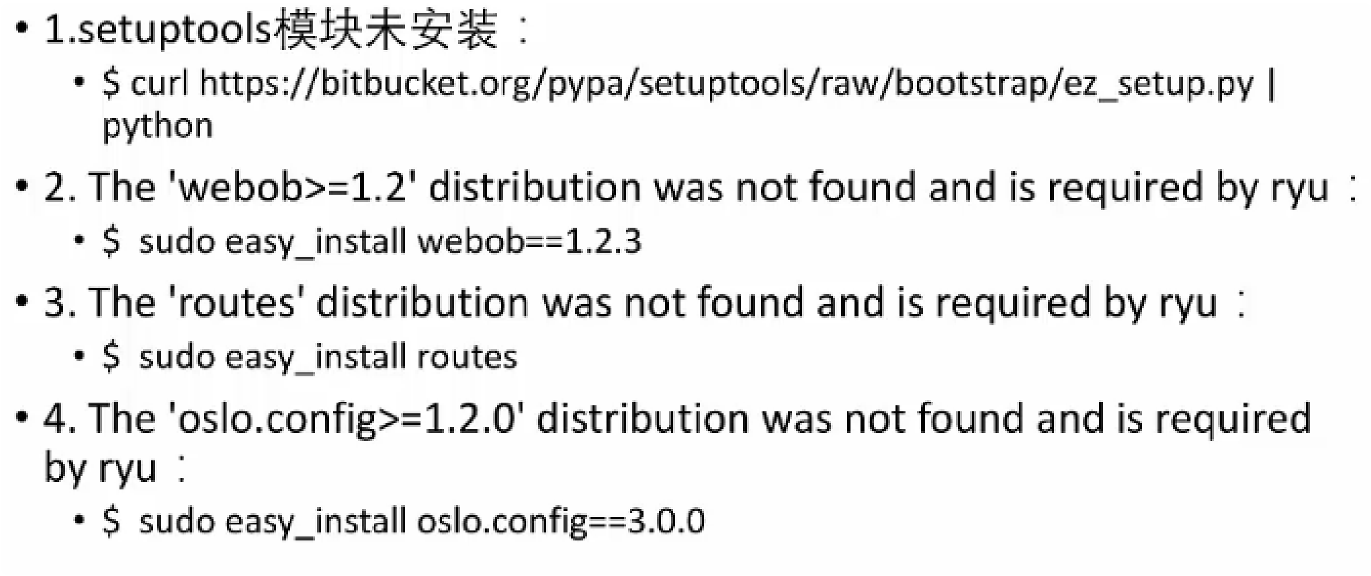
## （三）进行Ryu安装

Shell 复制代码

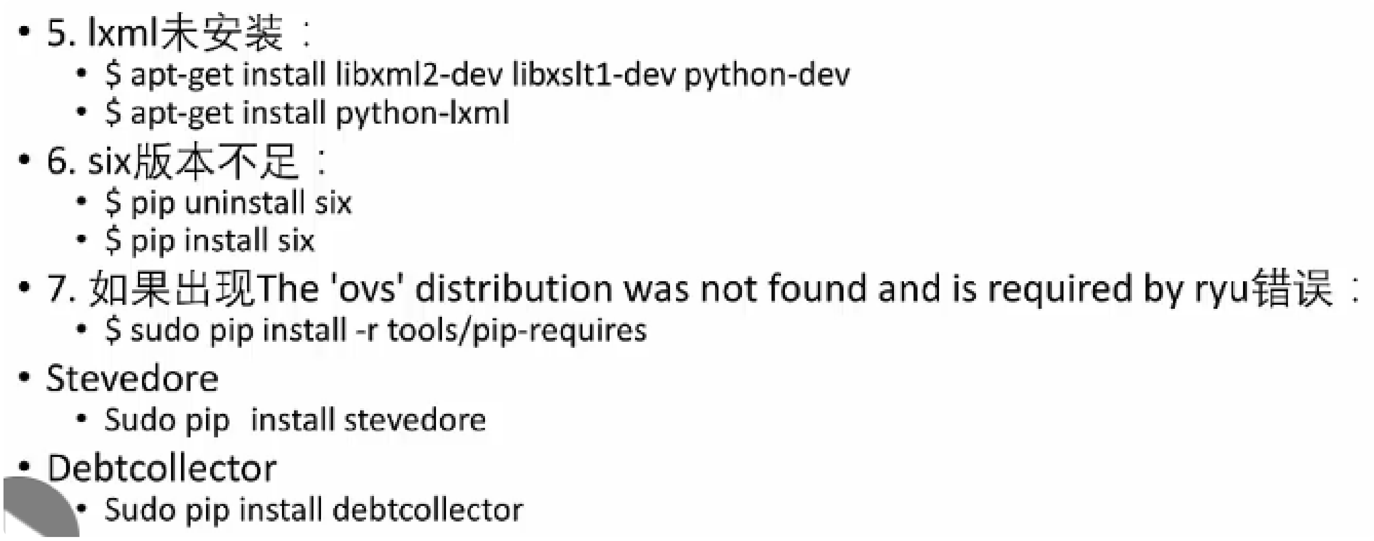
```
sudo python setup.py install
```

## （四）安装测试

### 1.常见问题及解决办法



- 1.setuptools模块未安装：
  - $ curl https://bitbucket.org/pypa/setuptools/raw/bootstrap/ez_setup.py | python
- 2. The 'webob>=1.2' distribution was not found and is required by ryu：
  - $  sudo easy_install webob==1.2.3
- 3. The 'routes' distribution was not found and is required by ryu：
  - $  sudo easy_install routes
- 4. The 'oslo.config>=1.2.0' distribution was not found and is required by ryu：
  - $  sudo easy_install oslo.config==3.0.0

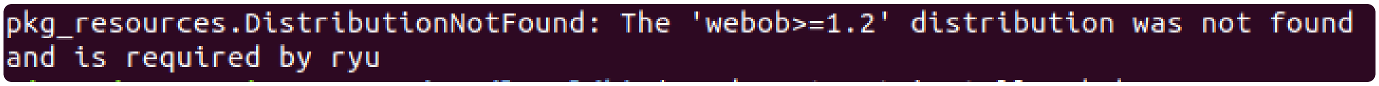


- 5. lxml未安装：
  - $ apt-get install libxml2-dev libxslt1-dev python-dev
  - $ apt-get install python-lxml
- 6. six版本不足：
  - $ pip uninstall six
  - $ pip install six
- 7. 如果出现The 'ovs' distribution was not found and is required by ryu错误：
  - $ sudo pip install -r tools/pip-requires
- Stevedore
  - Sudo pip  install stevedore
- Debtcollector
  - Sudo pip install debtcollector

### 2.安装测试

Shell 复制代码

```
njzy@njzy-Inspiron-5493:/usr/local/bin$ ryu-manager
```



```
pkg_resources.DistributionNotFound: The 'webob>=1.2' distribution was not found
and is required by ryu
```

```
njzy@njzy-Inspiron-5493:/usr/local/bin$ sudo easy_install webob==1.2.3
```

参考：

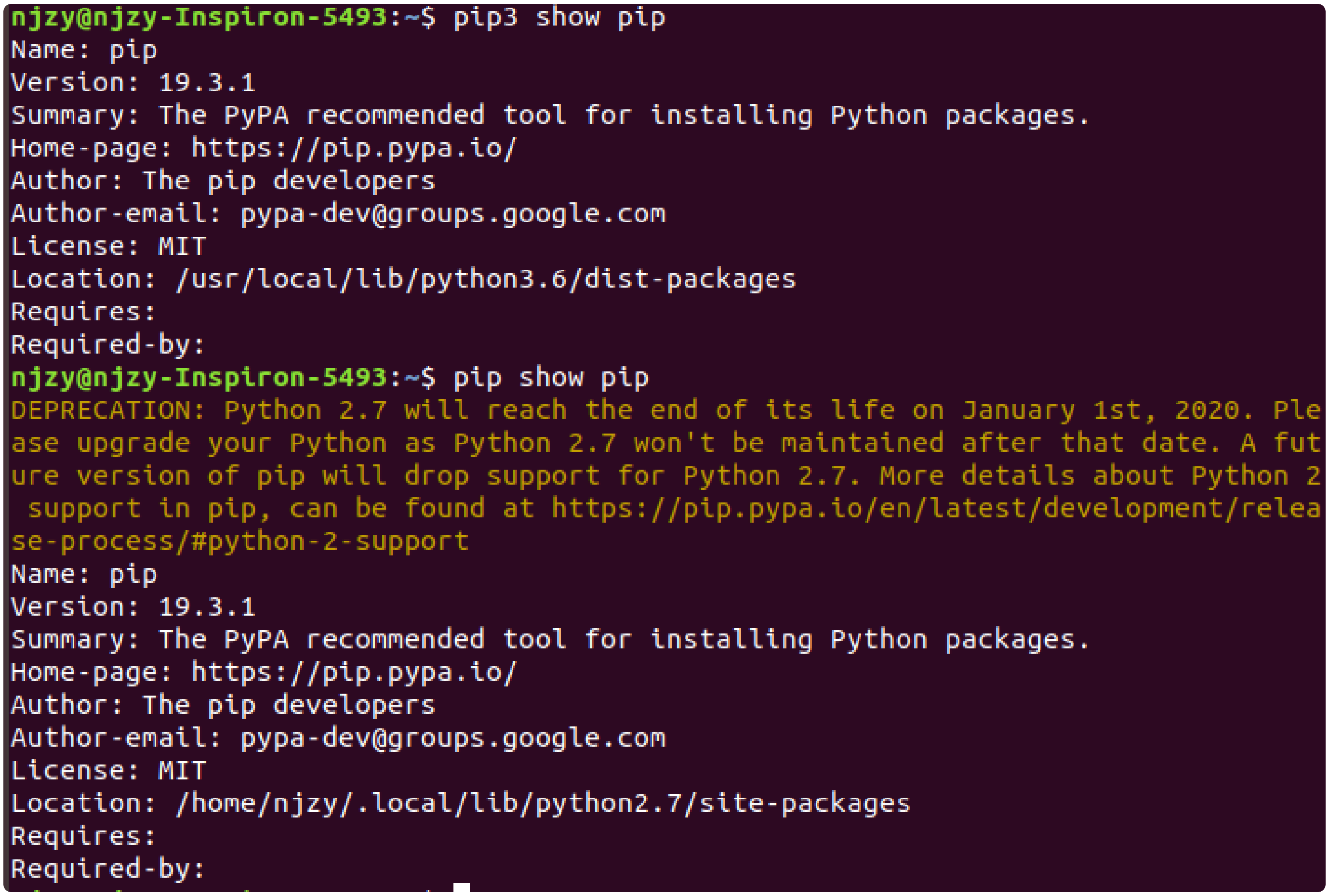
https://blog.csdn.net/cyz14/article/details/79994548

https://ryu-devel.narkive.com/AvVCZ4OS/ryu-installation-error-the-ovs-distribution-was-not-found-and-is-required-by-ryu

https://blog.csdn.net/u012424148/article/details/88951559

实际：

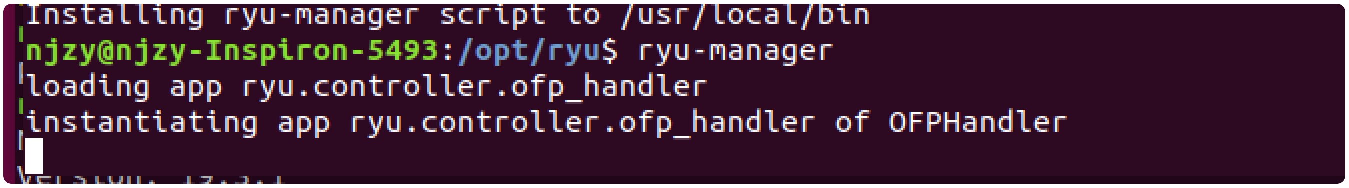
因为Ubuntu下，还含有一个Python3版本安装的pip3,所以我将依赖和安装中pip全部改成pip3，就安装成功了



```
njzy@njzy-Inspiron-5493:~$ pip3 show pip
Name: pip
Version: 19.3.1
Summary: The PyPA recommended tool for installing Python packages.
Home-page: https://pip.pypa.io/
Author: The pip developers
Author-email: pypa-dev@groups.google.com
License: MIT
Location: /usr/local/lib/python3.6/dist-packages
Requires:
Required-by:
njzy@njzy-Inspiron-5493:~$ pip show pip
DEPRECATION: Python 2.7 will reach the end of its life on January 1st, 2020. Ple
ase upgrade your Python as Python 2.7 won't be maintained after that date. A fut
ure version of pip will drop support for Python 2.7. More details about Python 2
 support in pip, can be found at https://pip.pypa.io/en/latest/development/relea
se-process/#python-2-support
Name: pip
Version: 19.3.1
Summary: The PyPA recommended tool for installing Python packages.
Home-page: https://pip.pypa.io/
Author: The pip developers
Author-email: pypa-dev@groups.google.com
License: MIT
Location: /home/njzy/.local/lib/python2.7/site-packages
Requires:
Required-by:
```

```shell
sudo pip3 install -r tools/pip-requires
sudo python3 setup.py install
ryu-manager
```

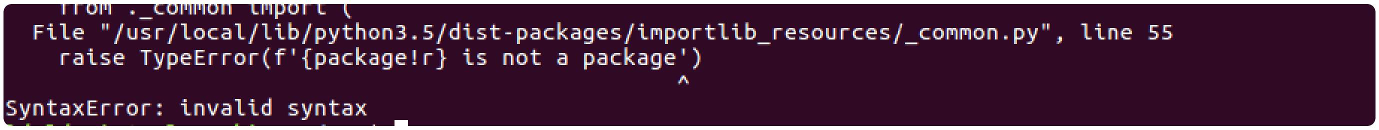
安装成功：



```
Installing ryu-manager script to /usr/local/bin
njzy@njzy-Inspiron-5493:/opt/ryu$ ryu-manager
loading app ryu.controller.ofp_handler
instantiating app ryu.controller.ofp_handler of OFPHandler
```

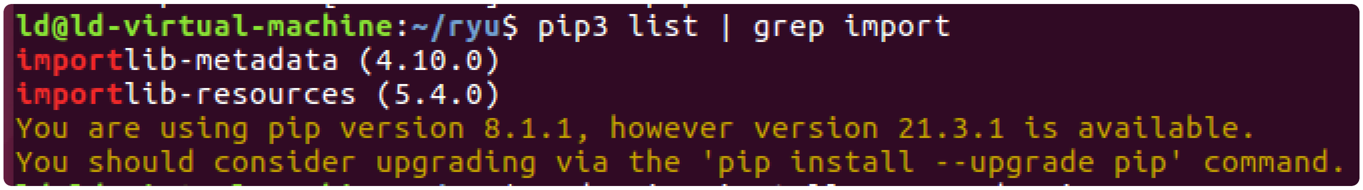
## （五）其他可能错误

### 1.importlib_resources中raise TypeError(f'{package!r} is not a package')



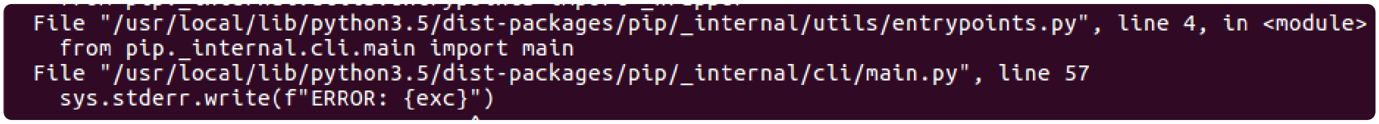
```
    from ._common import (
  File "/usr/local/lib/python3.5/dist-packages/importlib_resources/_common.py", line 55
    raise TypeError(f'{package!r} is not a package')
                                                  ^
SyntaxError: invalid syntax
```

先查看已经安装的packet：使用pip3 list即可



```
ld@ld-virtual-machine:~/ryu$ pip3 list | grep import
importlib-metadata (4.10.0)
importlib-resources (5.4.0)
You are using pip version 8.1.1, however version 21.3.1 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
```

发现提示更新，但是更新后使用pip可能出现以下问题：



```
  File "/usr/local/lib/python3.5/dist-packages/pip/_internal/utils/entrypoints.py", line 4, in <module>
    from pip._internal.cli.main import main
  File "/usr/local/lib/python3.5/dist-packages/pip/_internal/cli/main.py", line 57
    sys.stderr.write(f"ERROR: {exc}")
```

因此我们需要使用正确的pip版本（注：我们python版本为3.5.2）

```shell
wget https://bootstrap.pypa.io/pip/3.5/get-pip.py
python3 get-pip.py
```

之后卸载原有packet：

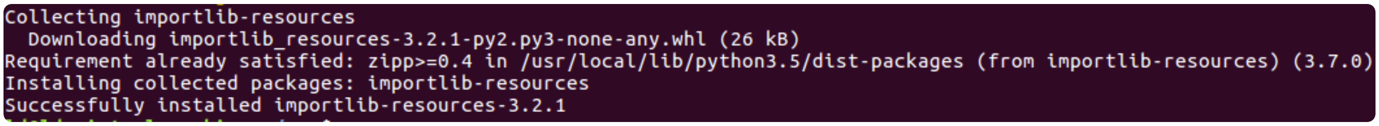
Shell 复制代码

```shell
sudo pip uninstall importlib-resources
```

进行重新安装：

Shell 复制代码

```shell
sudo pip install importlib-resources
```

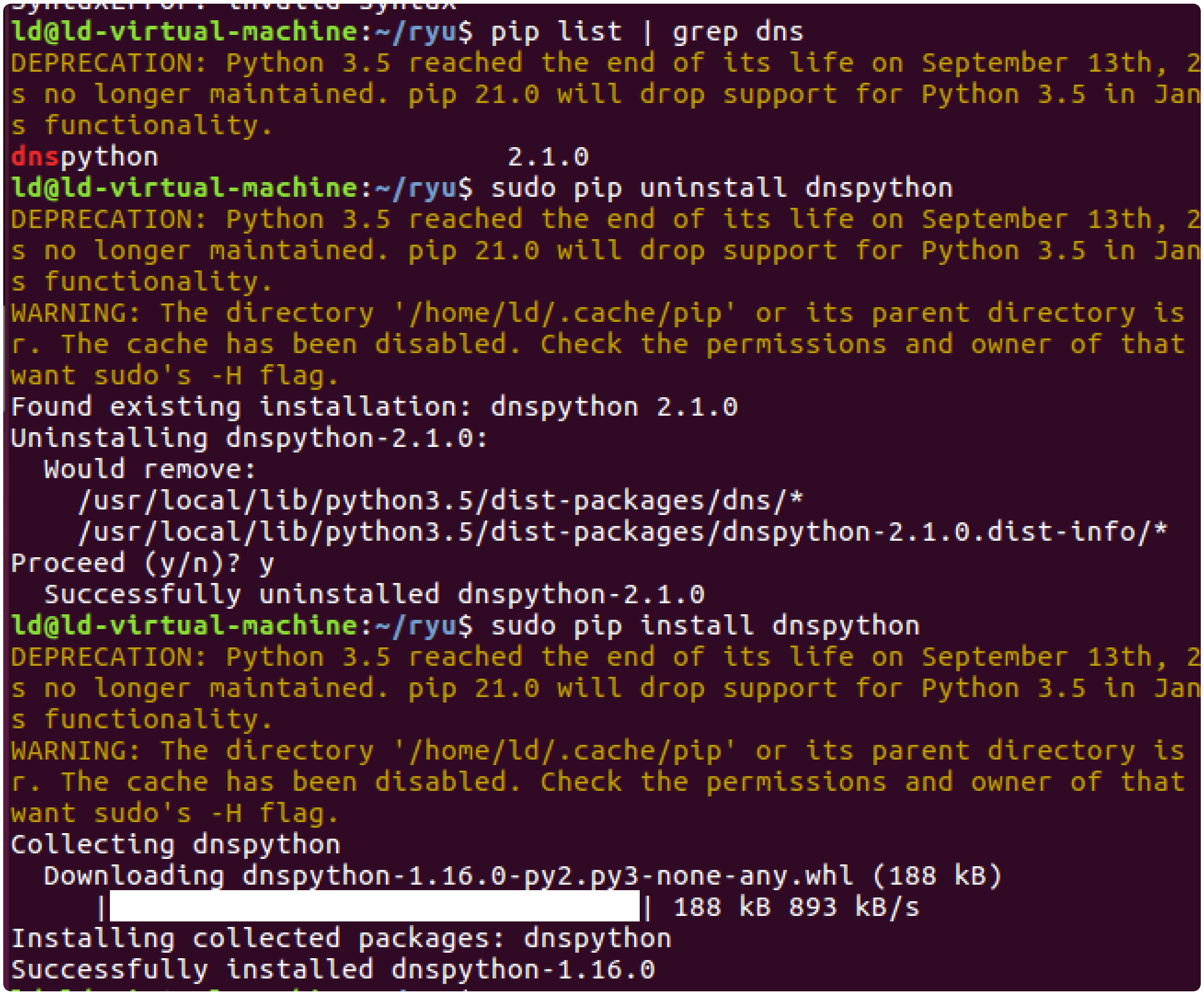


```
Collecting importlib-resources
  Downloading importlib_resources-3.2.1-py2.py3-none-any.whl (26 kB)
Requirement already satisfied: zipp>=0.4 in /usr/local/lib/python3.5/dist-packages (from importlib-resources) (3.7.0)
Installing collected packages: importlib-resources
Successfully installed importlib-resources-3.2.1
```

同样对于后面的importlib-metadata也是相同操作

## 2.raise NotImplementedError(f'unimplemented async backend {name}')

同上，先卸载，后安装：

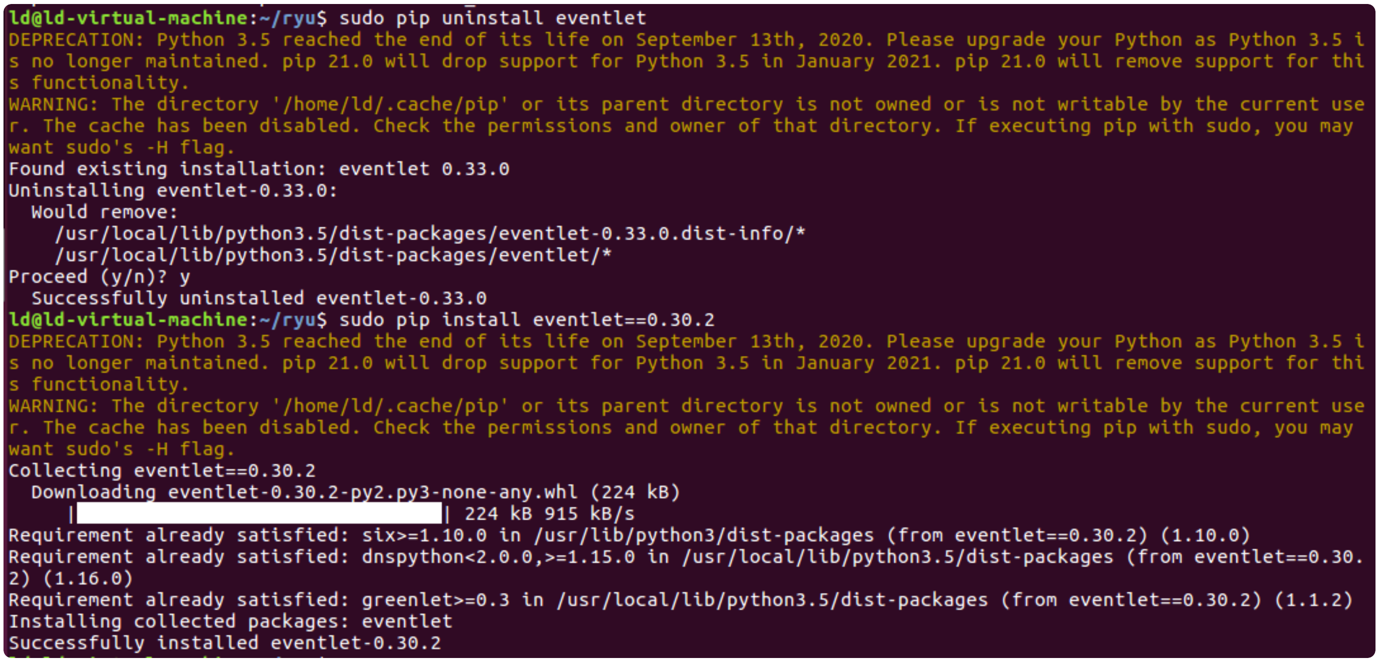


```
SyntaxError: invalid syntax
ld@ld-virtual-machine:~/ryu$ pip list | grep dns
DEPRECATION: Python 3.5 reached the end of its life on September 13th, 2
s no longer maintained. pip 21.0 will drop support for Python 3.5 in Jan
s functionality.
dnspython                    2.1.0
ld@ld-virtual-machine:~/ryu$ sudo pip uninstall dnspython
DEPRECATION: Python 3.5 reached the end of its life on September 13th, 2
s no longer maintained. pip 21.0 will drop support for Python 3.5 in Jan
s functionality.
WARNING: The directory '/home/ld/.cache/pip' or its parent directory is
r. The cache has been disabled. Check the permissions and owner of that
want sudo's -H flag.
Found existing installation: dnspython 2.1.0
Uninstalling dnspython-2.1.0:
  Would remove:
    /usr/local/lib/python3.5/dist-packages/dns/*
    /usr/local/lib/python3.5/dist-packages/dnspython-2.1.0.dist-info/*
Proceed (y/n)? y
  Successfully uninstalled dnspython-2.1.0
ld@ld-virtual-machine:~/ryu$ sudo pip install dnspython
DEPRECATION: Python 3.5 reached the end of its life on September 13th, 2
s no longer maintained. pip 21.0 will drop support for Python 3.5 in Jan
s functionality.
WARNING: The directory '/home/ld/.cache/pip' or its parent directory is
r. The cache has been disabled. Check the permissions and owner of that
want sudo's -H flag.
Collecting dnspython
  Downloading dnspython-1.16.0-py2.py3-none-any.whl (188 kB)
     |████████████████████████████████| 188 kB 893 kB/s
Installing collected packages: dnspython
Successfully installed dnspython-1.16.0
```

## 3.from eventlet.wsgi import ALREADY_HANDLED –-> ImportError: cannot import name 'ALREADY_HANDLED'

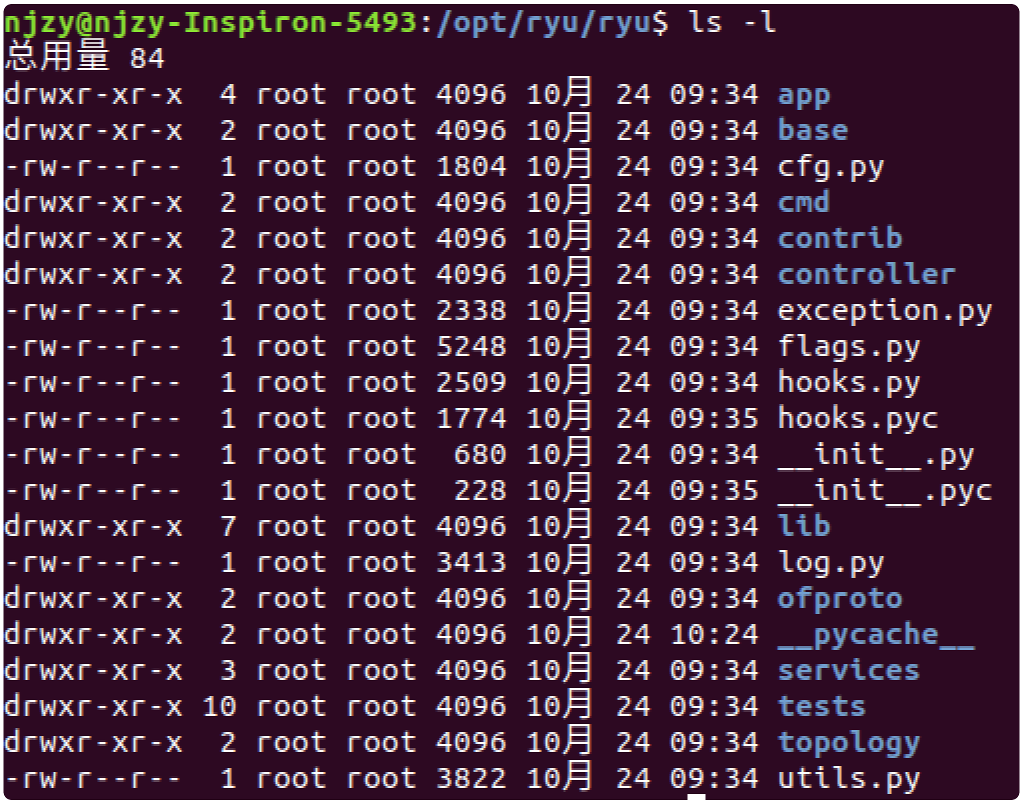
同样，先卸载，后安装，不过安装需要指定版本

```shell
pip3 install eventlet==0.30.2
```



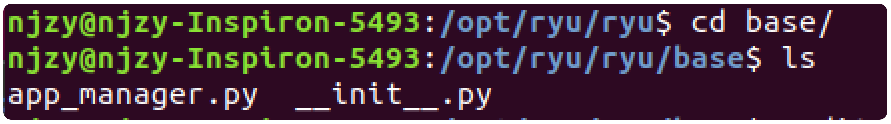
```
ld@ld-virtual-machine:~/ryu$ sudo pip uninstall eventlet
DEPRECATION: Python 3.5 reached the end of its life on September 13th, 2020. Please upgrade your Python as Python 3.5 i
s no longer maintained. pip 21.0 will drop support for Python 3.5 in January 2021. pip 21.0 will remove support for thi
s functionality.
WARNING: The directory '/home/ld/.cache/pip' or its parent directory is not owned or is not writable by the current use
r. The cache has been disabled. Check the permissions and owner of that directory. If executing pip with sudo, you may
want sudo's -H flag.
Found existing installation: eventlet 0.33.0
Uninstalling eventlet-0.33.0:
  Would remove:
    /usr/local/lib/python3.5/dist-packages/eventlet-0.33.0.dist-info/*
    /usr/local/lib/python3.5/dist-packages/eventlet/*
Proceed (y/n)? y
  Successfully uninstalled eventlet-0.33.0
ld@ld-virtual-machine:~/ryu$ sudo pip install eventlet==0.30.2
DEPRECATION: Python 3.5 reached the end of its life on September 13th, 2020. Please upgrade your Python as Python 3.5 i
s no longer maintained. pip 21.0 will drop support for Python 3.5 in January 2021. pip 21.0 will remove support for thi
s functionality.
WARNING: The directory '/home/ld/.cache/pip' or its parent directory is not owned or is not writable by the current use
r. The cache has been disabled. Check the permissions and owner of that directory. If executing pip with sudo, you may
want sudo's -H flag.
Collecting eventlet==0.30.2
  Downloading eventlet-0.30.2-py2.py3-none-any.whl (224 kB)
     |████████████████████████████████| 224 kB 915 kB/s
Requirement already satisfied: six>=1.10.0 in /usr/lib/python3/dist-packages (from eventlet==0.30.2) (1.10.0)
Requirement already satisfied: dnspython<2.0.0,>=1.15.0 in /usr/local/lib/python3.5/dist-packages (from eventlet==0.30.
2) (1.16.0)
Requirement already satisfied: greenlet>=0.3 in /usr/local/lib/python3.5/dist-packages (from eventlet==0.30.2) (1.1.2)
Installing collected packages: eventlet
Successfully installed eventlet-0.30.2
```

## 三：Ryu的源码目录



```
njzy@njzy-Inspiron-5493:/opt/ryu/ryu$ ls -l
总用量 84
drwxr-xr-x  4 root root 4096 10月  24 09:34 app
drwxr-xr-x  2 root root 4096 10月  24 09:34 base
-rw-r--r--  1 root root 1804 10月  24 09:34 cfg.py
drwxr-xr-x  2 root root 4096 10月  24 09:34 cmd
drwxr-xr-x  2 root root 4096 10月  24 09:34 contrib
drwxr-xr-x  2 root root 4096 10月  24 09:34 controller
-rw-r--r--  1 root root 2338 10月  24 09:34 exception.py
-rw-r--r--  1 root root 5248 10月  24 09:34 flags.py
-rw-r--r--  1 root root 2509 10月  24 09:34 hooks.py
-rw-r--r--  1 root root 1774 10月  24 09:35 hooks.pyc
-rw-r--r--  1 root root  680 10月  24 09:34 __init__.py
-rw-r--r--  1 root root  228 10月  24 09:35 __init__.pyc
drwxr-xr-x  7 root root 4096 10月  24 09:34 lib
-rw-r--r--  1 root root 3413 10月  24 09:34 log.py
drwxr-xr-x  2 root root 4096 10月  24 09:34 ofproto
drwxr-xr-x  2 root root 4096 10月  24 10:24 __pycache__
drwxr-xr-x  3 root root 4096 10月  24 09:34 services
drwxr-xr-x 10 root root 4096 10月  24 09:34 tests
drwxr-xr-x  2 root root 4096 10月  24 09:34 topology
-rw-r--r--  1 root root 3822 10月  24 09:34 utils.py
```

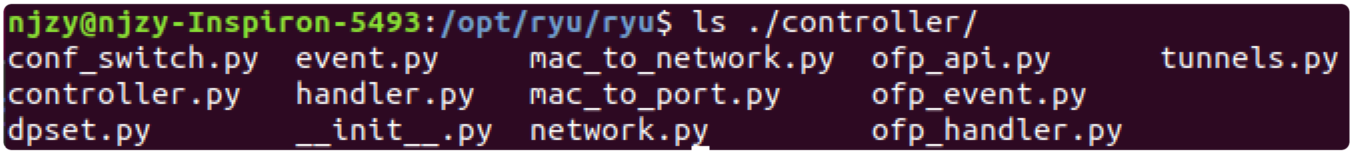
### （一）base



```
njzy@njzy-Inspiron-5493:/opt/ryu/ryu$ cd base/
njzy@njzy-Inspiron-5493:/opt/ryu/ryu/base$ ls
app_manager.py  __init__.py
```

base中有一个非常重要的文件：app_manager.py，其作用是RYU应用的管理中心。用于加载RYU应用程序，接受从APP发送过来的信息，同时也完成消息的路由。 其主要的函数有app注册、注销、查找、并定义了RYUAPP基类，定义了RYUAPP的基本属性。包含name, threads, events, event_handlers和observers等成员，以及对应的许多基本函数。如：start(), stop()等。 这个文件中还定义了AppManager基类，用于管理APP。定义了加载APP等函数。不过如果仅仅是开发APP的话，这个类可以不必关心。

## （二）controller

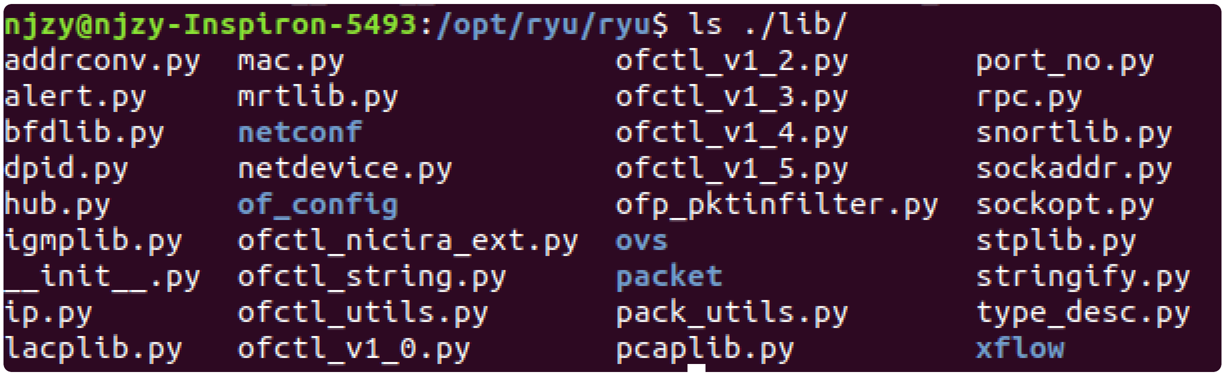


```
njzy@njzy-Inspiron-5493:/opt/ryu/ryu$ ls ./controller/
conf_switch.py  event.py     mac_to_network.py  ofp_api.py      tunnels.py
controller.py   handler.py   mac_to_port.py     ofp_event.py
dpset.py        __init__.py  network.py         ofp_handler.py
```

controller文件夹中许多非常重要的文件，如events.py, ofp_handler.py, controller.py等。其中controller.py中定义了OpenFlowController基类。用于定义OpenFlow的控制器，用于处理交换机和控制器的连接等事件，同时还可以产生事件和路由事件。其事件系统的定义，可以查看events.py和ofp_events.py。 在ofp_handler.py中定义了基本的handler句柄，完成了基本的如：握手，错误信息处理和keep alive 等功能。更多的如packet_in_handler应该在app中定义。 在dpset.py文件中，定义了交换机端的一些消息，如端口状态信息等，用于描述和操作交换机。如添加端口，删除端口等操作。

其他文件以后再叙述

## （三）lib

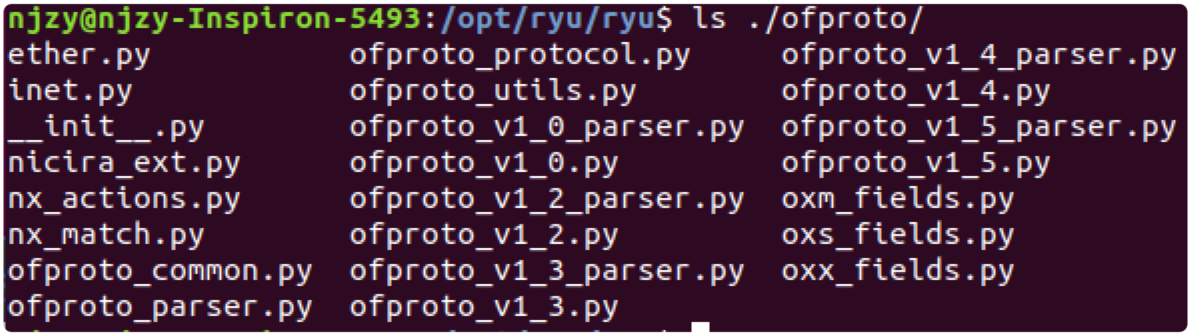


```
njzy@njzy-Inspiron-5493:/opt/ryu/ryu$ ls ./lib/
addrconv.py  mac.py               ofctl_v1_2.py       port_no.py
alert.py     mrtlib.py            ofctl_v1_3.py       rpc.py
bfdlib.py    netconf              ofctl_v1_4.py       snortlib.py
dpid.py      netdevice.py         ofctl_v1_5.py       sockaddr.py
hub.py       of_config            ofp_pktinfilter.py  sockopt.py
igmplib.py   ofctl_nicira_ext.py  ovs                 stplib.py
__init__.py  ofctl_string.py      packet              stringify.py
ip.py        ofctl_utils.py       pack_utils.py       type_desc.py
lacplib.py   ofctl_v1_0.py        pcaplib.py          xflow
```

lib中定义了我们需要使用到的基本的数据结构，如dpid, mac和ip等数据结构。在lib/packet目录下，还定义了许多网络协议，如ICMP, DHCP, MPLS和IGMP等协议内容。而每一个数据包的类中都有parser

和serialize两个函数。用于解析和序列化数据包。 lib目录下，还有ovs, netconf目录，对应的目录下有一些定义好的数据类型，不再赘述。

## （四）ofproto

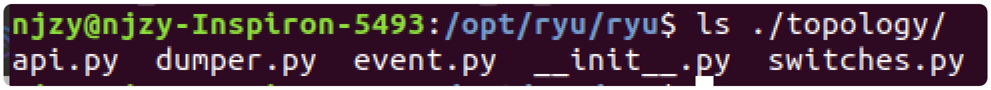


```
njzy@njzy-Inspiron-5493:/opt/ryu/ryu$ ls ./ofproto/
ether.py           ofproto_protocol.py     ofproto_v1_4_parser.py
inet.py            ofproto_utils.py        ofproto_v1_4.py
__init__.py        ofproto_v1_0_parser.py  ofproto_v1_5_parser.py
nicira_ext.py      ofproto_v1_0.py         ofproto_v1_5.py
nx_actions.py      ofproto_v1_2_parser.py  oxm_fields.py
nx_match.py        ofproto_v1_2.py         oxs_fields.py
ofproto_common.py  ofproto_v1_3_parser.py  oxx_fields.py
ofproto_parser.py  ofproto_v1_3.py
```

在这个目录下，基本分为两类文件，一类是协议的数据结构定义，另一类是协议解析，也即数据包处理函数文件。

如ofproto_v1_0.py是1.0版本的OpenFlow协议数据结构的定义，而ofproto_v1_0_parser.py则定义了1.0版本的协议编码和解码。具体内容不赘述，实现功能与协议相同。

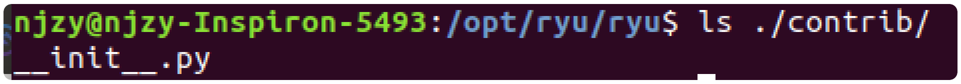
## （五）topology



```
njzy@njzy-Inspiron-5493:/opt/ryu/ryu$ ls ./topology/
api.py  dumper.py  event.py  __init__.py  switches.py
```

包含了switches.py等文件，基本定义了一套交换机的数据结构。 event.py定义了交换上的事件。 dumper.py定义了获取网络拓扑的内容。 最后api.py向上提供了一套调用topology目录中定义函数的接口。

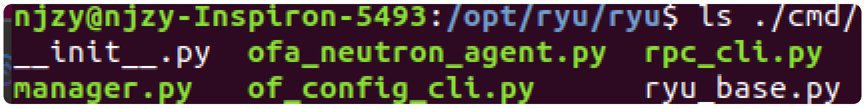
## （六）contrib



```
njzy@njzy-Inspiron-5493:/opt/ryu/ryu$ ls ./contrib/
__init__.py
```

这个文件夹主要存放的是开源社区贡献者的代码。

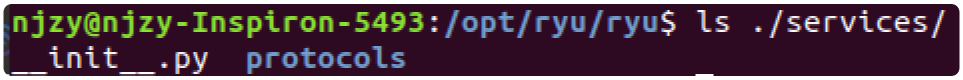
## （七）cmd



```
njzy@njzy-Inspiron-5493:/opt/ryu/ryu$ ls ./cmd/
__init__.py  ofa_neutron_agent.py  rpc_cli.py
manager.py   of_config_cli.py      ryu_base.py
```

定义了RYU的命令系统，具体不赘述。

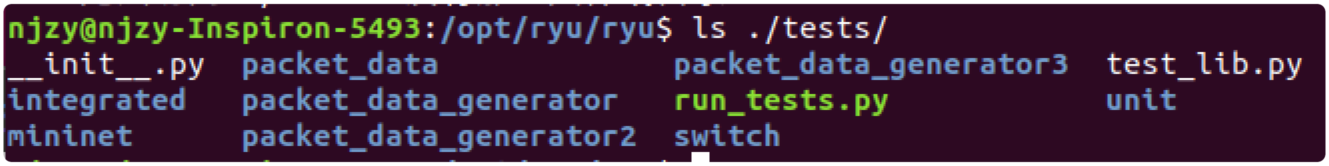
### （八）services



```
njzy@njzy-Inspiron-5493:/opt/ryu/ryu$ ls ./services/
__init__.py  protocols
```

完成了BGP和vrrp的实现。

### （九）tests



```
njzy@njzy-Inspiron-5493:/opt/ryu/ryu$ ls ./tests/
__init__.py  packet_data               packet_data_generator3  test_lib.py
integrated   packet_data_generator     run_tests.py            unit
mininet      packet_data_generator2    switch
```

tests目录下存放了单元测试以及整合测试的代码

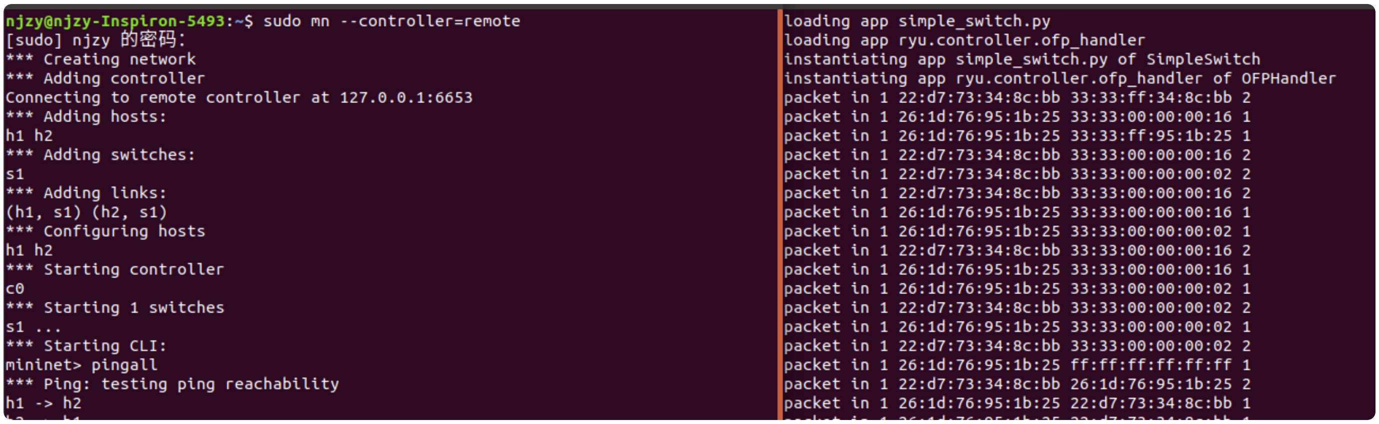
## 四：Ryu和Mininet联用

### （一）开启控制器，使用二层交换

```shell
~/ryu/ryu/app$ ryu-manager simple_switch.py
```

### （二）开启Mininet

```shell
sudo mn --controller=remote    //不指定拓扑--默认是一个交换机下面挂两个主机    --controller=remote  是将mininet的控制器指向远端控制器 未指定ip,默认0.0.0.0或者本地IP端口默认是在6635
```



```
njzy@njzy-Inspiron-5493:~$ sudo mn --controller=remote
[sudo] njzy 的密码:
*** Creating network
*** Adding controller
Connecting to remote controller at 127.0.0.1:6653
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet> pingall
*** Ping: testing ping reachability
h1 -> h2
```

```
loading app simple_switch.py
loading app ryu.controller.ofp_handler
instantiating app simple_switch.py of SimpleSwitch
instantiating app ryu.controller.ofp_handler of OFPHandler
packet in 1 22:d7:73:34:8c:bb 33:33:ff:34:8c:bb 2
packet in 1 26:1d:76:95:1b:25 33:33:00:00:00:16 1
packet in 1 26:1d:76:95:1b:25 33:33:ff:95:1b:25 1
packet in 1 22:d7:73:34:8c:bb 33:33:00:00:00:16 2
packet in 1 22:d7:73:34:8c:bb 33:33:00:00:00:02 2
packet in 1 22:d7:73:34:8c:bb 33:33:00:00:00:16 2
packet in 1 26:1d:76:95:1b:25 33:33:00:00:00:16 1
packet in 1 26:1d:76:95:1b:25 33:33:00:00:00:02 1
packet in 1 22:d7:73:34:8c:bb 33:33:00:00:00:16 2
packet in 1 26:1d:76:95:1b:25 33:33:00:00:00:16 1
packet in 1 26:1d:76:95:1b:25 33:33:00:00:00:02 1
packet in 1 22:d7:73:34:8c:bb 33:33:00:00:00:02 2
packet in 1 26:1d:76:95:1b:25 33:33:00:00:00:02 1
packet in 1 22:d7:73:34:8c:bb 33:33:00:00:00:02 2
packet in 1 26:1d:76:95:1b:25 ff:ff:ff:ff:ff:ff 1
packet in 1 22:d7:73:34:8c:bb 26:1d:76:95:1b:25 2
packet in 1 26:1d:76:95:1b:25 22:d7:73:34:8c:bb 1
```