

Setting up the Ryu Controller

Introduction

A professor of mine that's guided me contacted me today, and asked me as to whether I was free enough to teach some of my juniors on how to set up SDN controllers.

Considering that it's been a while and that just setting up simple remote controllers like Ryu shouldn't take up too much of my time, I figured, why the hell not?

So, I agreed. And, then remembered that I reset my laptop's windows side after Apex Legends kept crashing. That meant that I lost all my configurations and Virtual Machines that I set up on VirtualBox.

It might've been a pain if my goal was to showcase my project, since I'd have to set up everything from scratch, but considering that I need to show the guys on how to set stuff up from the ground anyway, this works out.

Setting up the workspace

In order to make this easy for everyone, I decided to switch back to Windows and used the VMWare Workstation Player software to set up a Virtual Machine. (I've used VirtualBox and the number of errors it's thrown at me is enough to try an alternative)

Then, I downloaded an Ubuntu ISO onto my system, and attempted to set that up in VMWare. If someone wants to follow along to this article, here's links to both of them.

- [VMWare Workstation Player](#)
- [Ubuntu 22.04.2 LTS ISO](#)

Now, it was time to set it up on VMware Workstation. The steps to do so are,

1. Create a New Virtual Machine.
2. Provide the location of the ISO in the Installer disc image file.
3. Provide credentials (testuser, test)
4. Name the VM (ryu_controller)
5. Provide disk space (20gb, but honestly up to you)
6. Wait for a bit while VMware easyinstall does the magic for you.
7. Go through the installation procedure. (minimal install)
8. Make the user again? (testuser, test)
9. Wait for a bit
10. Restart
11. Login

↑

Dependency Hell

Well, now that we have the workspace (VM) setup, it's time to get Ryu set up. It might be tempting to directly hit the `pip install ryu` command and call it but note that there's two issues with this,

1. The package has major dependency conflicts.
2. The python version support only runs upto version 3.9

That being said, let's get started. Open a terminal and update the source list, so that you can install packages with ease.

```
sudo apt-get update && sudo apt-get upgrade
sudo apt-get install gcc libffi-dev libssl-dev libxml2-dev libxslt1-dev
mkdir Code && cd Code
sudo apt-get install curl git vim
curl https://pyenv.run | bash
```



We'll have to do something a little scuffed now, we need to set up pyenv so that we can change the version of our python easily. In order to activate pyenv properly, append this at the end of your `~/home/<username>/.bashrc`.

```
export PYENV_ROOT="$HOME/.pyenv"
command -v pyenv >/dev/null || export PATH="$PYENV_ROOT/bin:$PATH"
eval "$(pyenv init -)"
```

Then, do the following,

```
source .bashrc

# Install python3.9.6
pyenv install 3.9.6

# Create a virtualenv for python3.9.6
pyenv virtualenv 3.9.6 ryu
pyenv activate ryu

# Clone and install the dependencies for the controller
git clone https://github.com/faucetsdn/ryu.git
cd ryu
pip install .

# Start the manager to see what kind of output you get.
ryu-manager
```

If you've done everything correctly, you should end up with the following STDOUT:

```
loading app ryu.controller.ofp_handler
instantiating app ryu.controller.ofp_handler of OFPHandler
```

If that's the screen you're ending up at, you're doing great, because you've just managed to set up the Ryu controller for yourself!

"Do note that Ryu is not maintained by it's creators anymore, and that does also mean that the support is super low now."

Running examples

While it may catch your eye pretty fast, I had some trouble understanding the documentation and how to operate with Ryu the first time that I worked with it.

So, to reduce time and effort in finding the already existing examples, here's the path to it.

```
cd /home/<username>/Code/Ryu/ryu/ryu/app/  
ryu-manager <script>.py
```

This should be enough to experiment with different kinds of controller programs.

Moving on?

Ideally, you should connect this to the mininet setup that you have set up. If you're a little confused or need a refresher on how to work with Mininet, you can check out another article that I [wrote about that](#).