

Wiki for Seting up Routing Recovering  
and  
Bandwidth Management System Based on Zynq

Maolin Wang

October 4, 2014

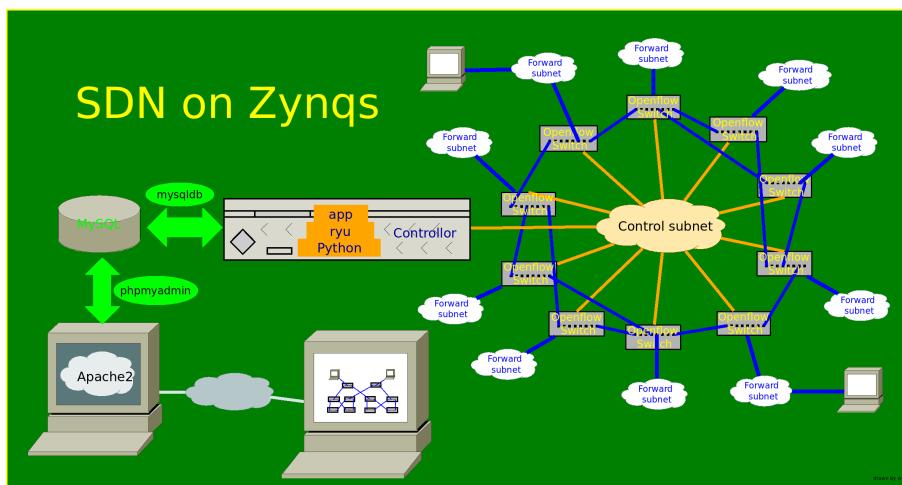
# 1 Overview

## 1.1 Brief Introduction to SDN

SDN(Software Defined Network) is a new network architecture which is booming in both academic and industry world. The most import conception of SDN is the separation of control layer and forwarding layer, which make the behavior of the network programmable. More infomation about SDN can be found on ONF's website.<sup>1</sup>

## 1.2 Routing Recovery And Bindwidth Management

As the programability of SDN, researchers can build many useful applications based on SDN architecture. Professor Chengchen Hu of Xi'an Jitong University and his team implement a routing recovering and bandwidth management system based on Zynq. The system chart is shown as the following picture:



First switches supporting OpenFlow Protocol 1.3<sup>2</sup> are implemented on Zeds. Then a PC serving as the controller together with several openflow switches forms the software defined network. Finally, Routing recovering from network breakdown and bindwidth management are implemented in this network.

## 1.3 Contents of this wiki Document

This document will interprets details about procedure of setting up the system. More details about the design and implementation of the system can be found on official website of OpenHW.<sup>3</sup>

---

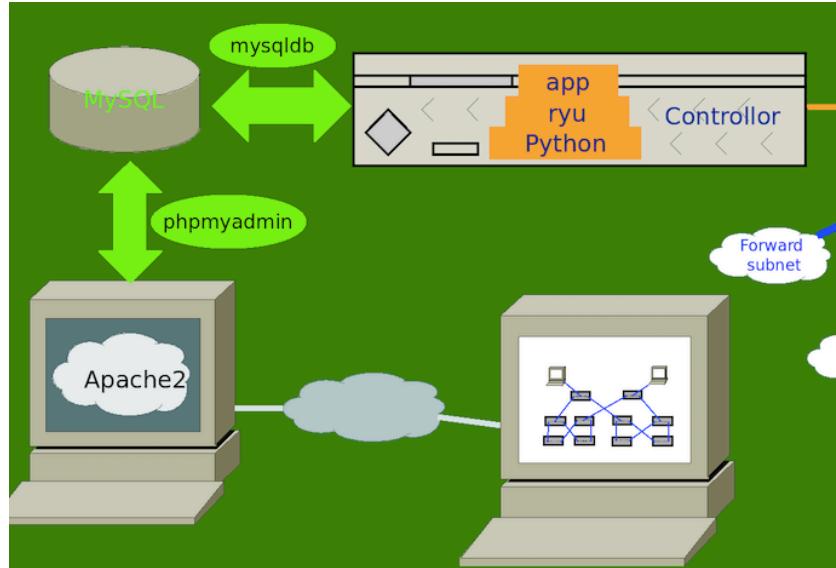
<sup>1</sup>[www.opennetworking.org](http://www.opennetworking.org)

<sup>2</sup>protocol used by controller and forwarding device to communicate with each other

<sup>3</sup><http://www.openhw.org/project/index.php?act=viewid=2579>

## 2 Controller

A PC installed Ubuntu 14.04 is used to serve as the controller of the SDN.



This computer is used to run the controller program, store information about the network into mysql database and distribute the information through a HTTP server. Material needed to install ryu controller can be found in the folder ryu, Other software can be installed by apt-get. The following subsections will give the guidance for the configuration of the controller.

### 2.1 Brief Introduction to Ryu

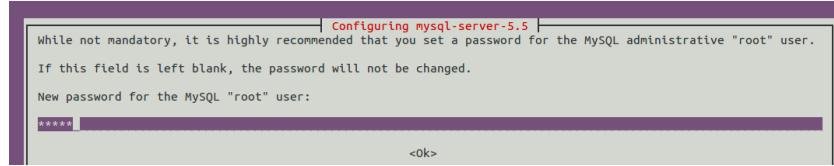
Controller serves as the manager of network. And there are many open-source implementations of openflow controller on Linux such as NOX, POX, Floodlight and Beacon. In this Project ryu is used. As ryu is written in python, python 2.7 is installed on the PC. More info about ryu can be found on its official website.

### 2.2 Database Setup

Install mysql

```
xup@xilinx-E6500:~/Desktop$ sudo apt-get install mysql-server mysql-client
```

config the password of mysql database to be mysql.



**Configuration** create database we need

```
mpg@mpg:~/Desktop/new_sdn$ mysql -u root -p
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 5.5.35-0ubuntu0.14.04.1 (Ubuntu)

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> use new_sdn;
Database changed
mysql> select * from user;
Query OK, 0 rows affected (0.00 sec)
Records: 0  Duplicates: 0  Warnings: 0
Query OK, 0 rows affected (0.00 sec)
Query OK, 0 rows affected (0.00 sec)
Records: 0  Duplicates: 0  Warnings: 0
Query OK, 0 rows affected (0.00 sec)
Query OK, 0 rows affected (0.00 sec)
Query OK, 0 rows affected (0.00 sec)
```

check the table in database meshsr

```
mysql> show tables;
+-----+
| Tables_in_meshsr |
+-----+
| flowEntry          |
| meshsr_connection |
| meshsr_node        |
| phyLink            |
| ports              |
| serverNIC          |
| switches           |
+-----+
7 rows in set (0.01 sec)
```

## 2.3 Python-mysqldb Setup

install python module named `mysqldb`

```
xup@xilinx-E6500:~/Desktop$ sudo apt-get install python-mysqldb
```

## 2.4 Website Setup

## install apache2

```
xup@xilinx-E6500:~/Desktop$ sudo apt-get install apache2
```

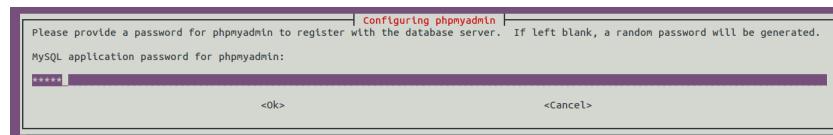
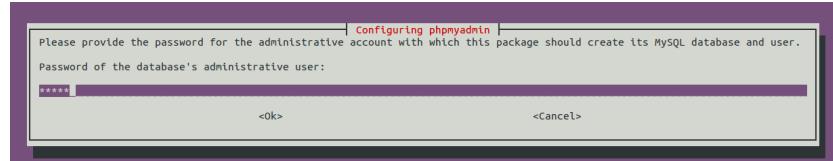
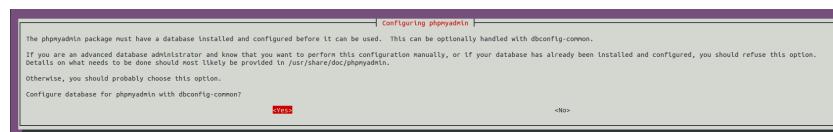
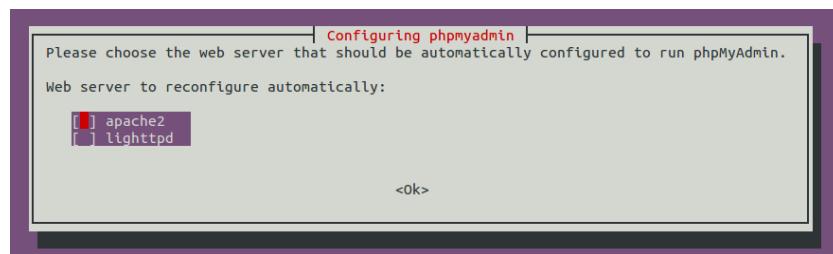
install php5 and libapache2-mod-php5

```
xup@xilinx-E6500:~/Desktop$ sudo apt-get install php5 libapache2-mod-php5
```

install myphpadmin

```
xup@xilinx-E6500:~/Desktop$ sudo apt-get install phpmyadmin
```

configure phpmyadmin



Configuration of website

```

root@xilinx-E6500:~/Desktop
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    ServerName localhost
    DocumentRoot /var/www/meshsr
    <Directory /var/www/meshsr>
        Require all granted
        AddHandler mod_php5 .php
        Options Indexes Followsymlinks
        AllowOverride None
        DirectoryIndex index.php
    </Directory>

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf
</VirtualHost>
```

```
xup@xilinx-E6500:~/Desktop/new_sdn$ sudo cp -r meshsr /var/www/
```

```
root@xilinx-E6500:/var/www# chmod -R 755 meshsr/
```

```

root@xilinx-E6500:/var/www# /etc/init.d/apache2 restart
* Restarting web server apache2
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
[ OK ]
```

## 2.5 Ryu Setup

install ryu

```
xup@xilinx-E6500:~/Desktop/new_sdn/ryu$ sudo apt-get install python-pip
```

```
xup@xilinx-E6500:~/Desktop/new_sdn/ryu$ sudo python setup.py install
```

install some missing modules of python

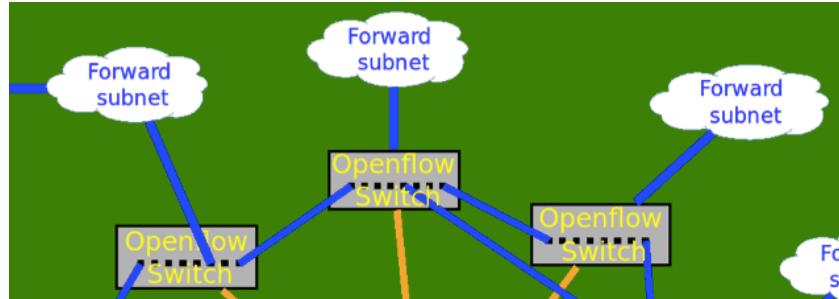
```
root@xilinx-E6500:/home# ryu-manager -v
Traceback (most recent call last):
  File "/usr/local/bin/ryu-manager", line 5, in <module>
    from pkg_resources import load_entry_point
  File "/usr/lib/python2.7/dist-packages/pkg_resources.py", line 2749, in <module>
    working_set = WorkingSet._build_master()
  File "/usr/lib/python2.7/dist-packages/pkg_resources.py", line 444, in _build_master
    ws.require(_requires_)
  File "/usr/lib/python2.7/dist-packages/pkg_resources.py", line 725, in require
    needed = self.resolve(parse_requirements(requirements))
  File "/usr/lib/python2.7/dist-packages/pkg_resources.py", line 628, in resolve
    raise DistributionNotFound(req)
pkg_resources.DistributionNotFound: repoze.lru>=0.3
```

```
root@xilinx-E6500:/home# sudo apt-cache search repo
python-repoze.lru - tiny LRU cache implementation and decorator
python3-repoze.lru - tiny LRU cache implementation and decorator for Python 3
python-pyramid - Pyramid web application framework, a Pylons project
python-repoze.sphinx.autointerface - Sphinx extension that auto-generates API docs from Zope interfaces
python-repoze.tm2 - Zope-like transaction manager via WSGI middleware
python-repoze.what - authorization framework for Python WSGI applications
python-repoze.what-plugins - authorization framework for Python WSGI applications - plugins collection
python-repoze.who - identification and authentication framework for Python WSGI applications
python-repoze.who-plugins - authentication framework for Python WSGI applications - plugins collection
python-translationstring - Utility library for i18n relied on by various Repoze packages - Python 3.x
python3-translationstring - Utility library for i18n relied on by various Repoze packages - Python 2.x
```

```
xup@xilinx-E6500:~$ sudo apt-get install python-repoze.lru
```

### 3 Switches

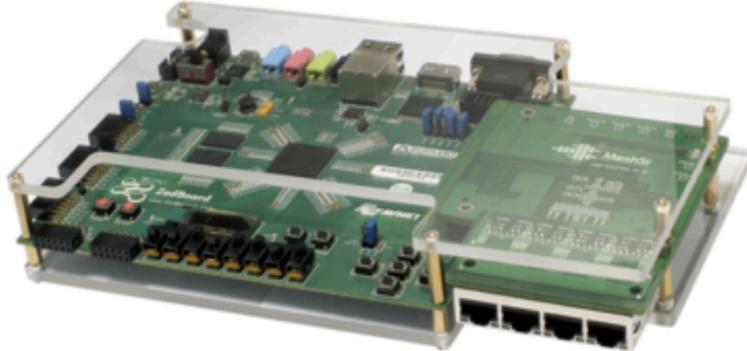
Switches are on the forwarding layer of the network, which use Openflow Protocol to communicate with the controller.



Main function of these switches are forwarding packets according to the flowtable.

#### 3.1 ONetSwitch20

In this project, ONetSwitch20 is chosen as the platform to implement openflow switch, which consists of a Zedboard and a FMC card with 4 ethernet ports. Details about software and hardware design of the switches can be found in the paperwork of the project.



ONetSwitch is a product of MeshSr Co. Ltd. More information about the corporation and the product can be found on its website.<sup>4</sup>

#### 3.2 Configuration

**Make Boot SD cards** The switches are set to boot from SD card. To do that, we need to partition the SD card and copy files to each partition.

---

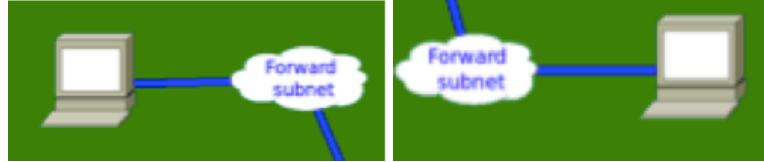
<sup>4</sup><http://www.meshsr.com>

- partition the SD card into two partition labeled with boot and fs.
- copy boot materials(BOOT.bin,devicetree.dtb,uImage) to boot partition.
- use "rsync -a" to copy ext4 filesystem to fs partition.

**IP,MAC address and dpid modification** To avoid collision, we should modify the IP and MAC address of each switch after the switch boots. Besides, the datapathid(dpid) of each switch should also be different. The method is to log into the switch through console, and change the configurations in the file /mnt/init.sh.

## 4 Terminals

In this demo, two terminals used to generating and receiving video stream are connected to the network.



Functions of routing recovering and bandwidth management can be observed visually through the change of the video played on the terminal which is in charge of receiving data.

### 4.1 Configuration

In this demonstration, two PC installed Ubuntu 14.04 are used as the two terminals. And VLC is used to generate and receive the video stream. To install VLC on ubuntu 14.04, the typical steps are:

- sudo add-apt-repository ppa:videolan/master-daily
- sudo apt-get update
- sudo apt-get install vlc

## 5 System Setup

In this section the procedure to set up a SDN with 5 switches is shown. The final system built is shown as the following picture.



### 5.1 Switches Setup

For the convenience of reference, we note that the serial numbers of the Zed are 0, 1, 2, 3, 4 from up to down. And the number of each Zed ethernet ports are 1, 2, 3, 4 from left to right.

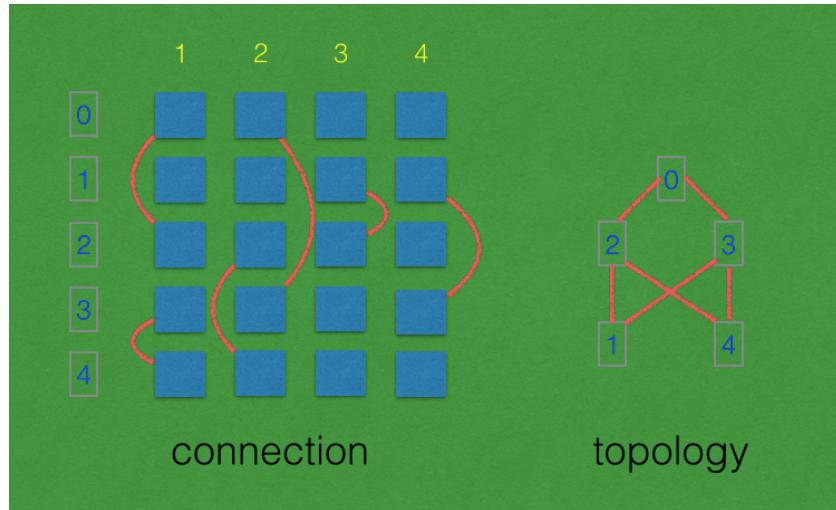


Set the IP address of the 5 switches to 10.0.0.10-14 and the dpid to 10-14. And

Don't forget to make sure that MAC address of each ethernet port are different.

### 5.1.1 Cable Connection

A typical connection of the switches is shown in the following picture.



## 5.2 Terminals Setup

One terminal serves as the video server and the other serves as a client. Call the video server S for short, and call the video client C for short.

### 5.2.1 Boot

After finishing booting, choose PC1 to serve as C and choose PC2 to serve as S.

### 5.2.2 IP Configuration

Open a terminal(type **ctrl+alt T**) on **S**, change to root user, the command is

**sudo -s**

And then run command:

```
ifconfig eth0 up  
(if this doesn't work ,change eth0 to eth1)  
ifconfig eth0 192.168.1.101  
ping 192.168.1.100
```

Open a terminal(type **ctrl+alt T**) on **C**, change to root user, the command is

**sudo -s**

And then run command:

```
ifconfig eth0 up  
(if this doesn't work ,change eth0 to eth1)  
ifconfig eth0 192.168.1.100  
ping 192.168.1.101
```

### 5.3 Controller Setup

Before running the controller program, Make sure all the Zed should have been booted. Then connect controller PC and the Zed PS RJ45 port to a switch. Then connect the S(PC2) to number 0(on the bottom) FMC card RJ45 Then connect C(PC1) to number 3(the 4th one) FMC card RJ45.

*Note:Every time you rerun the demo, the Zed should reset.*

#### 5.3.1 IP Configuration

Open a terminal on **Controller PC**(type **ctrl+alt T**) and config the IP to 10.0.0.1, the command is

```
ifconfig eth0 up  
(if this doesn't work ,change eth0 to eth1)  
ifconfig eth0 10.0.0.1
```

#### 5.3.2 Run the Demo

Go to the ryu home directory and run the commands:

```
./topo_stop.sh  
./cleanup_db.sh  
./topo_start.sh
```

wait for the 12 links appear.(about 60 seconds)

ping each other until success.

open the browser, and check network topology on website localhost/meshsr/index.php

#### 5.3.3 Note

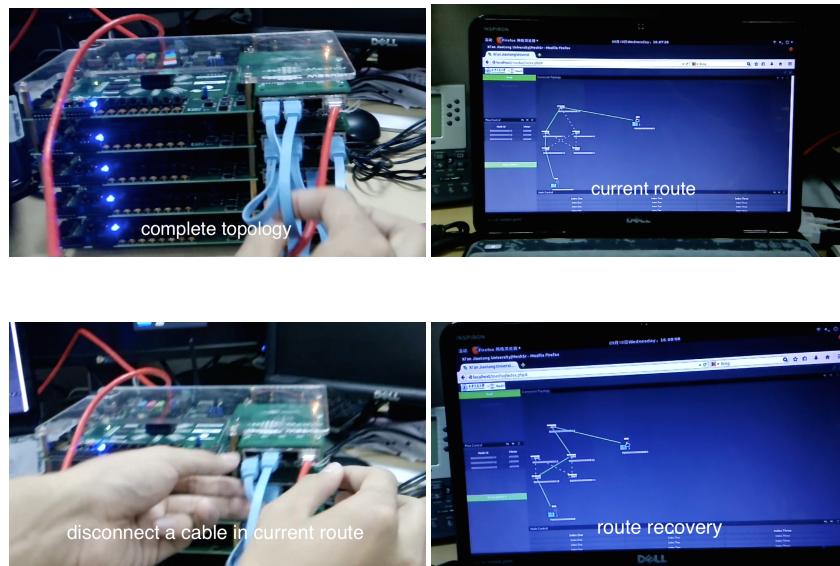
If ping won't success, we need to bind the mac address and ip address of server and client manually. First, open terminals both on C(PC1) and S(PC2), find the MAC address of S(PC2) by using command ifconfig on the terminal of S(PC1). Then config the static ARP table on C by using command:

```
arp -s x.x.x.x XX:XX:XX:XX:XX:XX
```

x.x.x.x is IP of S and XX:XX:XX:XX:XX:XX is the MAC of S.

## 5.4 Route Recovering

After disconnect a link, the ping won't stop. And after 10 seconds the change can be observed on the website by refreshing. The following pictures show routing recovering.



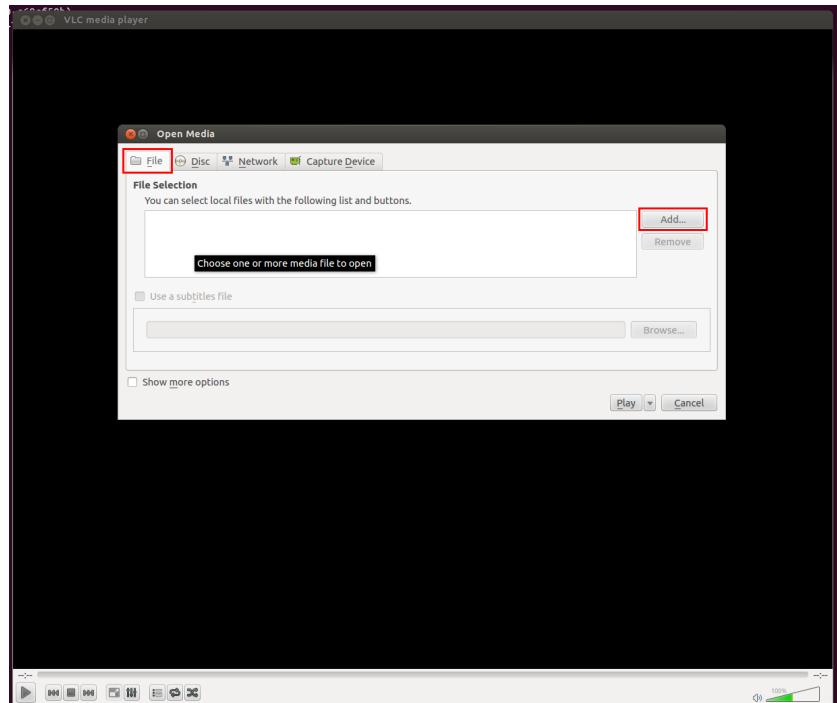
## 5.5 Video Streaming

### 5.5.1 Stream Generating

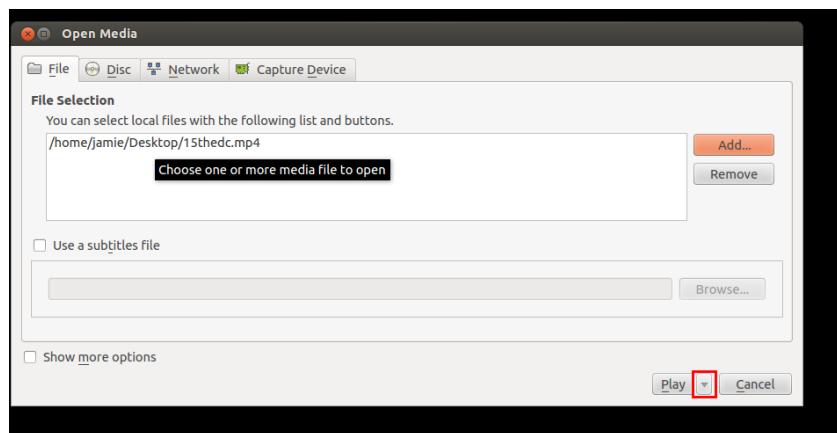
Open a terminal(type **ctrl+alt T**) on S, run command:

```
vlc
```

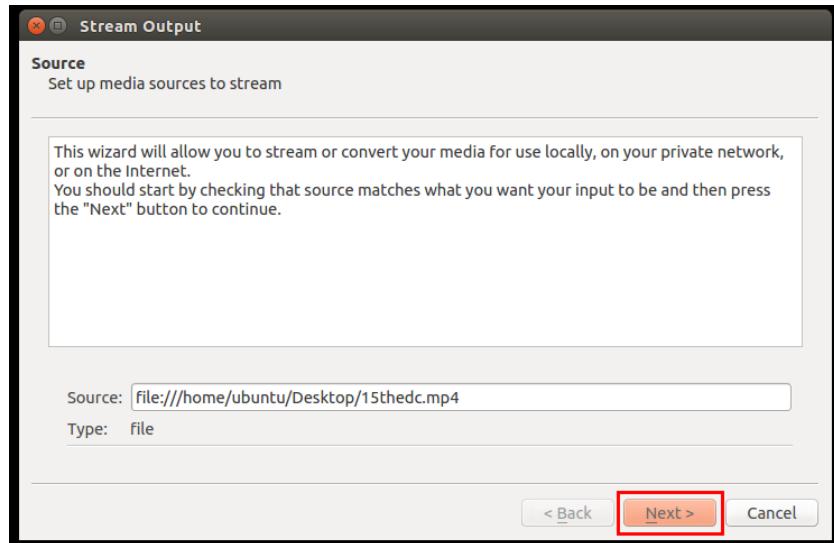
to open the vlc player. Right click on vlc and choose open Media → open file



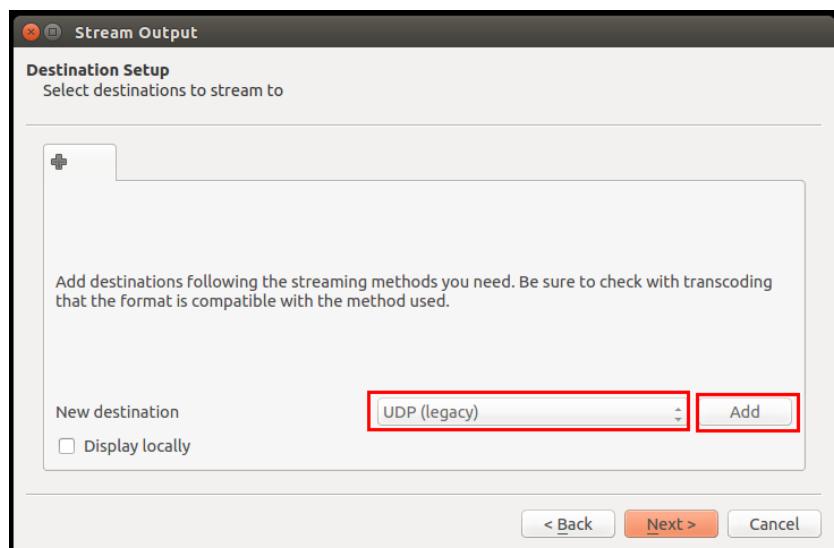
click Add and add a video file. Then click open.



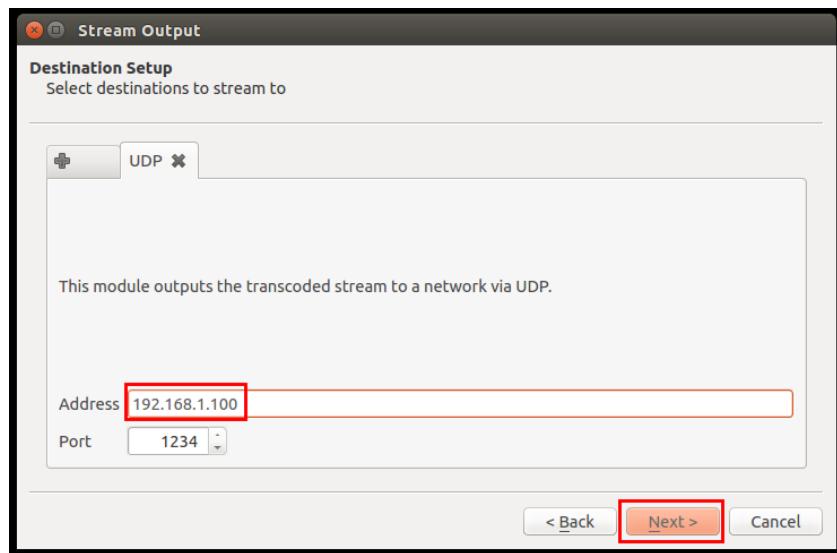
type **alt+s** to stream the video.



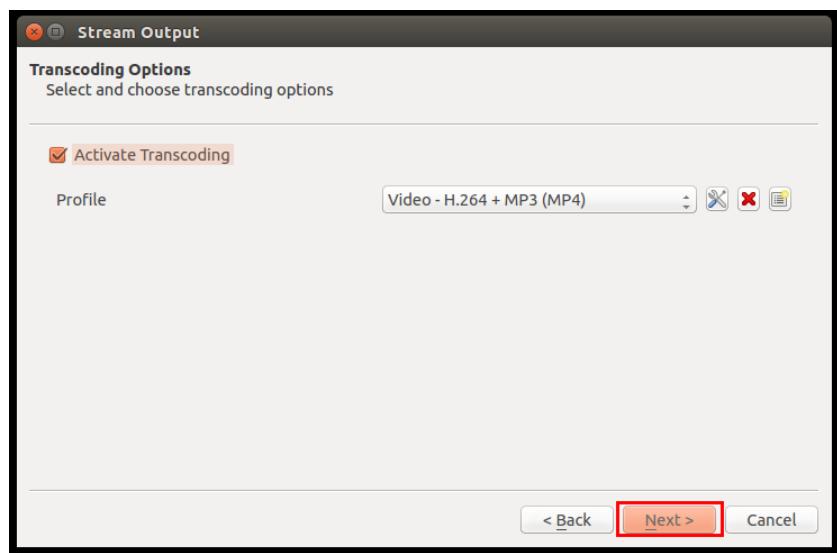
click next



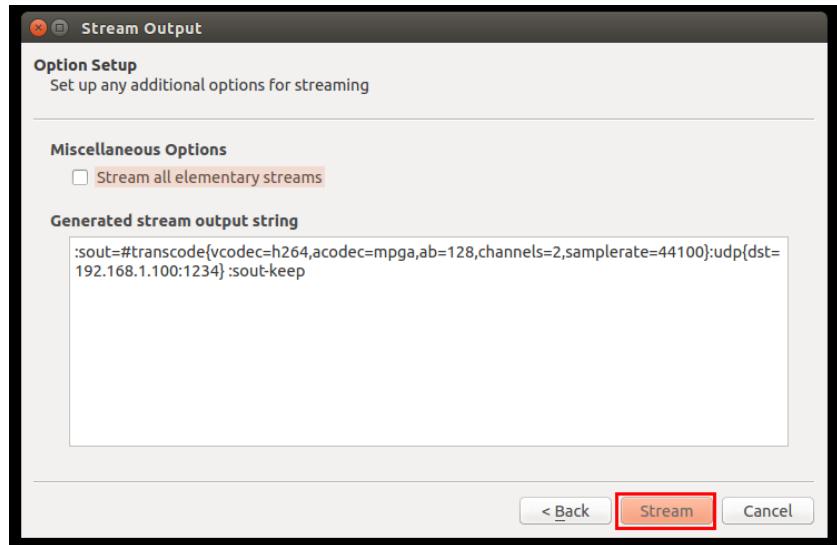
choose UDP(legacy) and click Add



Type in the IP address of S.click next



click next



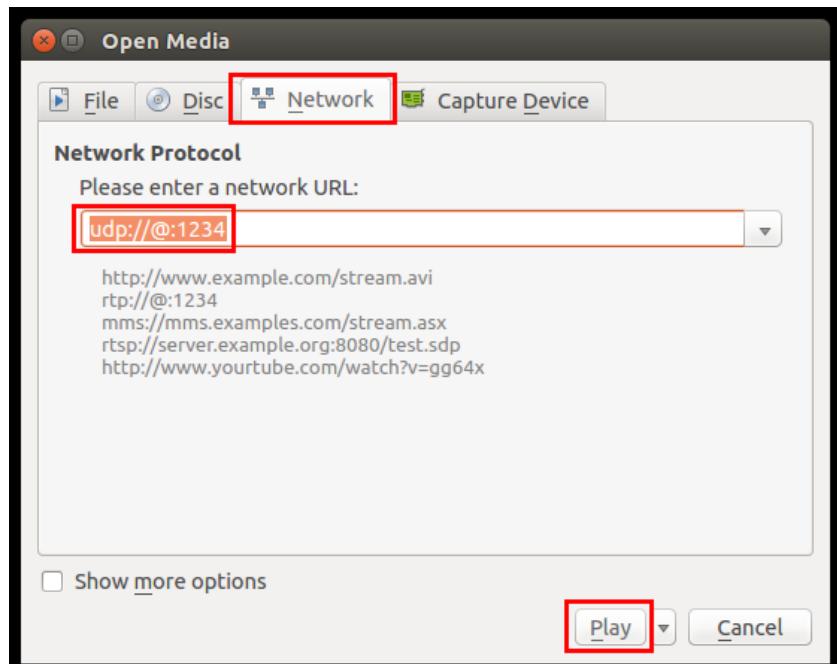
click stream

### 5.5.2 Stream Receiving

Open a terminal(type **ctrl+alt T**) on **C**, run command:

**vlc**

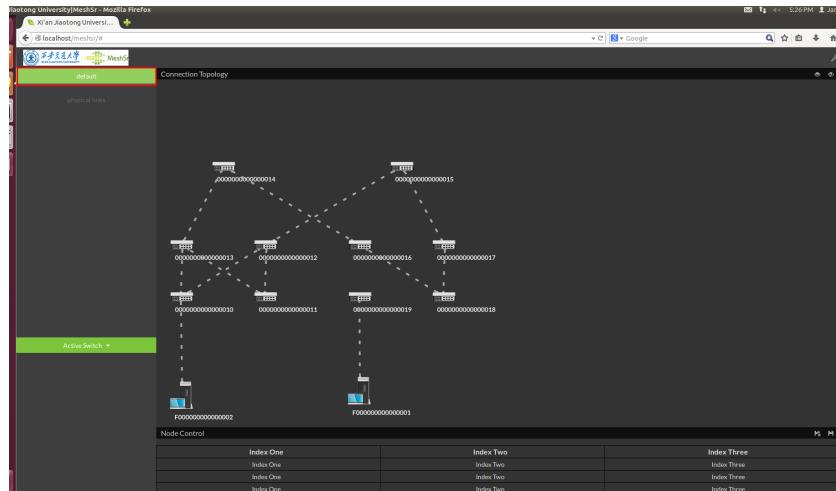
to open the vlc player. Right click on vlc and choose open Media→ open Network



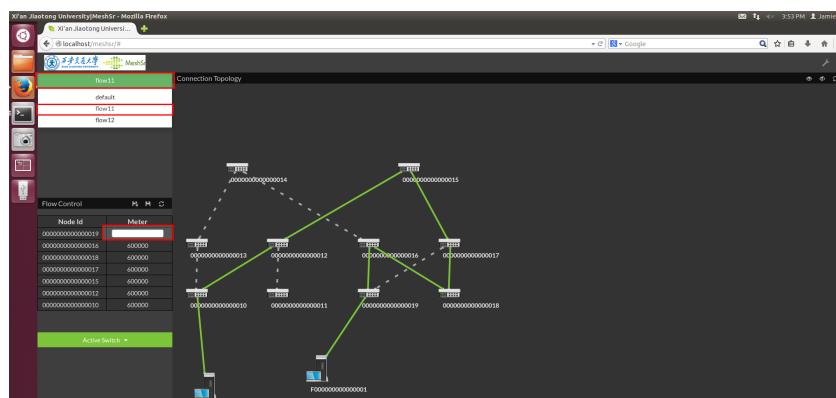
type in `udp://@:1234`, then click play.

## 5.6 Traffic Limitation

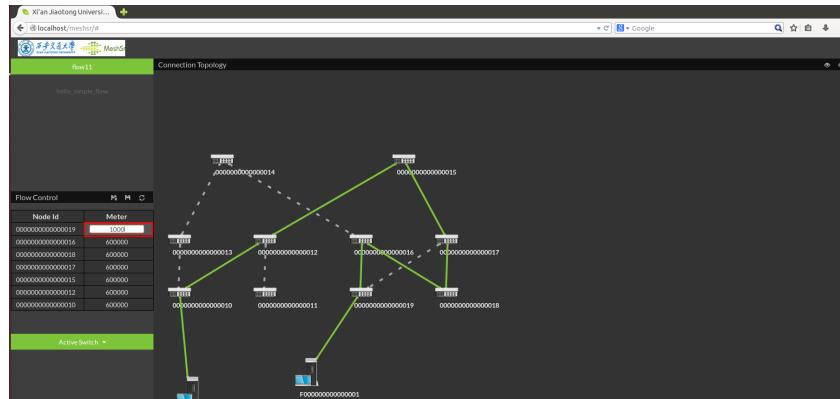
click 'default' up left of the browser.



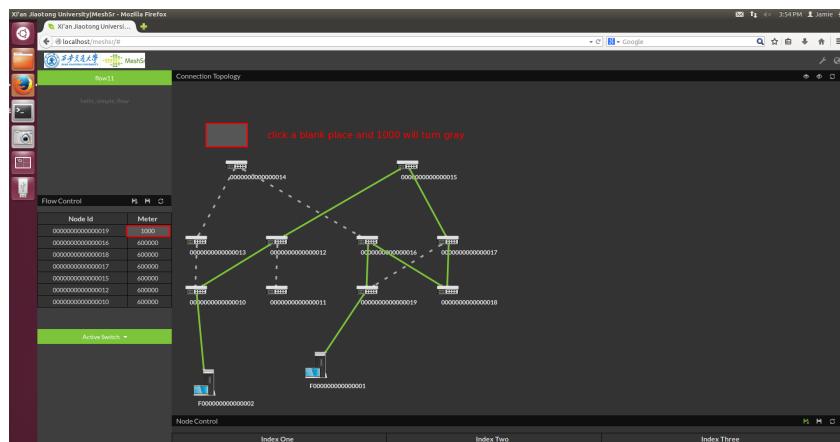
choose flowx, click one of the 'Meter' column in Flow Control .



Type in the 1000(or less).



click a blank place then click save symbol.



Then the effect of traffic limitation can be observed.