Lab2

翁牧言 521030910007

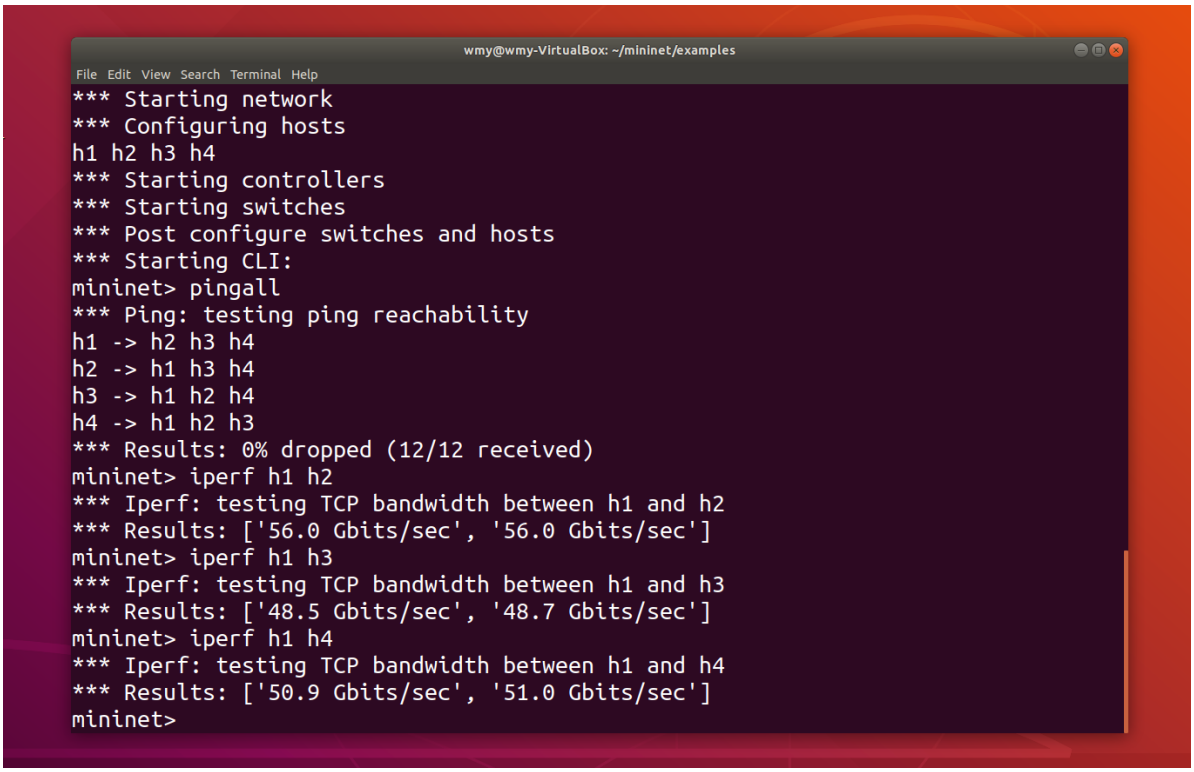
1

hw1.py

```
#!/usr/bin/env python

from mininet.net import Mininet
from mininet.node import Controller, RemoteController, OVSController
from mininet.node import CPULimitedHost, Host, Node
from mininet.node import OVSKernelSwitch, UserSwitch
from mininet.node import IVSSwitch
from mininet.cli import CLI
from mininet.log import setLogLevel, info
from mininet.link import TCLink, Intf
from subprocess import call

def myNetwork():

    net = Mininet( topo=None,
                   build=False,
                   ipBase='10.0.0.0/8')

    info( '*** Adding controller\n' )
    c0=net.addController(name='c0',
                         controller=RemoteController,
                         ip='127.0.0.1',
                         protocol='tcp',
                         port=6633)

    info( '*** Add switches\n')
    s1 = net.addSwitch('s1', cls=OVSKernelSwitch, dpid='0000000000000001')
    s2 = net.addSwitch('s2', cls=OVSKernelSwitch, dpid='0000000000000002')
    s3 = net.addSwitch('s3', cls=OVSKernelSwitch, dpid='0000000000000003')

    info( '*** Add hosts\n')
    h1 = net.addHost('h1', cls=Host, ip='10.0.0.1', defaultRoute=None)
    h2 = net.addHost('h2', cls=Host, ip='10.0.0.2', defaultRoute=None)
    h3 = net.addHost('h3', cls=Host, ip='10.0.0.3', defaultRoute=None)
    h4 = net.addHost('h4', cls=Host, ip='10.0.0.4', defaultRoute=None)

    info( '*** Add links\n')
    net.addLink(h1, s1)
    net.addLink(s1, s2, bw=20)
    net.addLink(s2, h2)
    net.addLink(h4, s1)
    net.addLink(s1, s3, bw=20)
```

```
    net.addLink(h3, s3)

    info( '*** Starting network\n')
    net.build()
    info( '*** Starting controllers\n')
    for controller in net.controllers:
        controller.start()

    info( '*** Starting switches\n')
    net.get('s1').start([c0])
    net.get('s2').start([c0])
    net.get('s3').start([c0])

    info( '*** Post configure switches and hosts\n')

    CLI(net)
    net.stop()

if __name__ == '__main__':
    setLogLevel( 'info' )
    myNetwork()
```

The iperf3 outputs



2

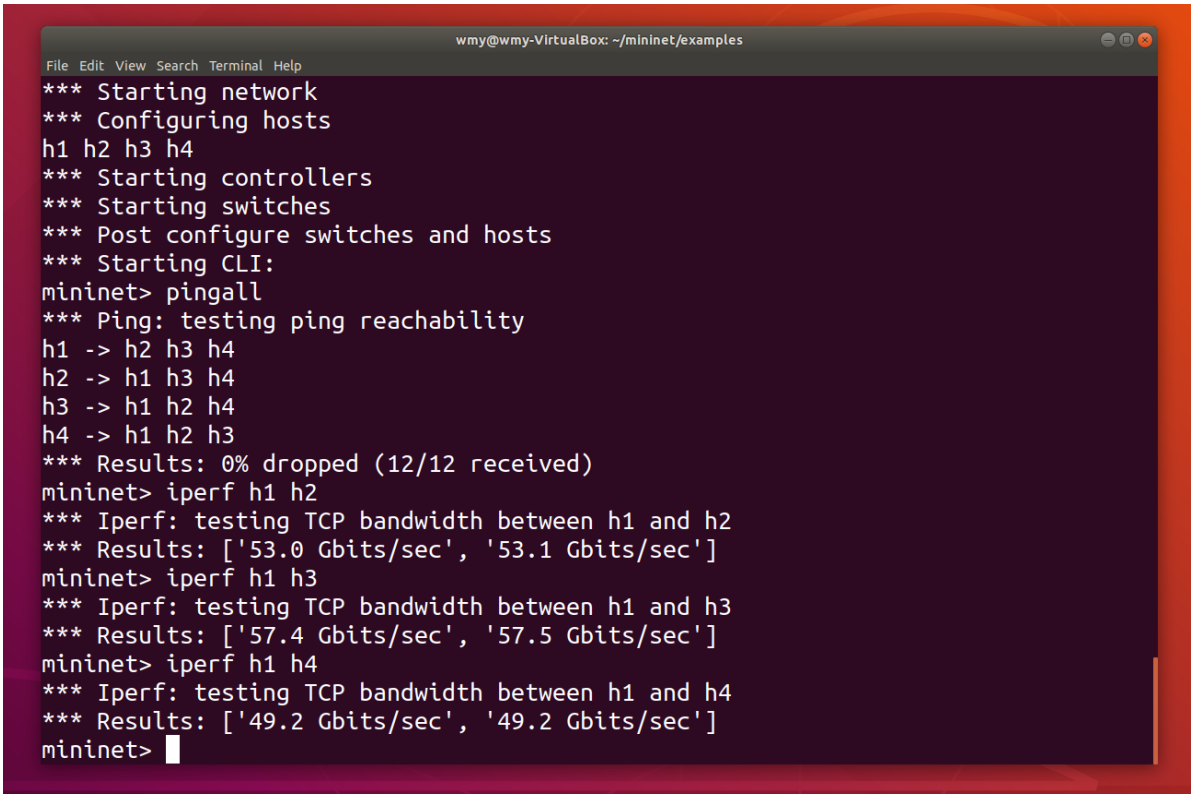
hw2.py

```
#!/usr/bin/env python

from mininet.net import Mininet
from mininet.node import Controller, RemoteController, OVSController
```

```
from mininet.node import CPULimitedHost, Host, Node
from mininet.node import OVSKernelSwitch, UserSwitch
from mininet.node import IVSSwitch
from mininet.cli import CLI
from mininet.log import setLogLevel, info
from mininet.link import TCLink, Intf
from subprocess import call

def myNetwork():

    net = Mininet( topo=None,
                   build=False,
                   ipBase='10.0.0.0/8')

    info( '*** Adding controller\n' )
    c0=net.addController(name='c0',
                      controller=RemoteController,
                      ip='127.0.0.1',
                      protocol='tcp',
                      port=6633)

    info( '*** Add switches\n')
    s1 = net.addSwitch('s1', cls=OVSKernelSwitch, dpid='0000000000000001')
    s2 = net.addSwitch('s2', cls=OVSKernelSwitch, dpid='0000000000000002')
    s3 = net.addSwitch('s3', cls=OVSKernelSwitch, dpid='0000000000000003')

    info( '*** Add hosts\n')
    h1 = net.addHost('h1', cls=Host, ip='10.0.0.1', defaultRoute=None)
    h2 = net.addHost('h2', cls=Host, ip='10.0.0.2', defaultRoute=None)
    h3 = net.addHost('h3', cls=Host, ip='10.0.0.3', defaultRoute=None)
    h4 = net.addHost('h4', cls=Host, ip='10.0.0.4', defaultRoute=None)

    info( '*** Add links\n')
    net.addLink(h1, s1)
    net.addLink(s1, s2, bw=20, loss=5)
    net.addLink(s2, h2)
    net.addLink(h4, s1)
    net.addLink(s1, s3, bw=20, loss=5)
    net.addLink(h3, s3)

    info( '*** Starting network\n')
    net.build()
    info( '*** Starting controllers\n')
    for controller in net.controllers:
        controller.start()

    info( '*** Starting switches\n')
    net.get('s1').start([c0])
    net.get('s2').start([c0])
    net.get('s3').start([c0])

    info( '*** Post configure switches and hosts\n')

    CLI(net)
    net.stop()
```

```
if __name__ == '__main__':
    setLogLevel( 'info' )
    myNetwork()
```

the iperf3 outputs

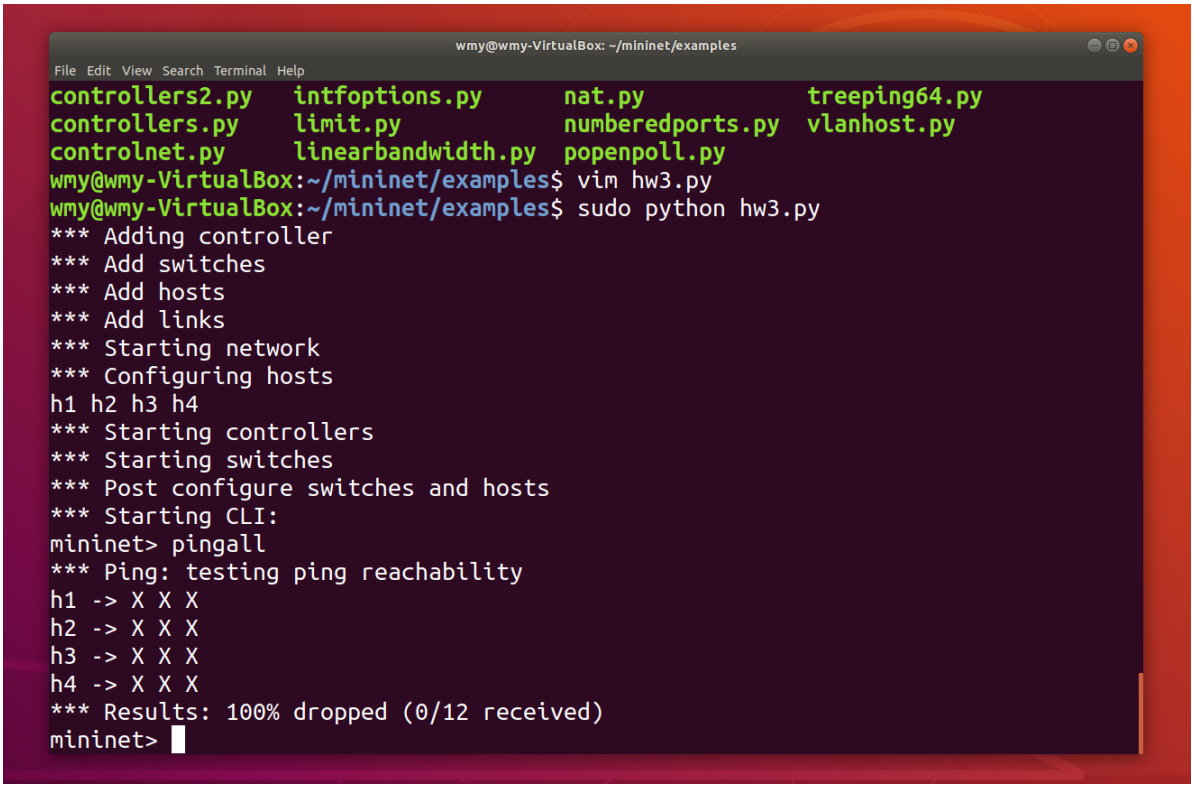


3

```
#!/usr/bin/env python

from mininet.net import Mininet
from mininet.node import Controller, RemoteController, OVSController
from mininet.node import CPULimitedHost, Host, Node
from mininet.node import OVSKernelSwitch, UserSwitch
from mininet.node import IVSSwitch
from mininet.cli import CLI
from mininet.log import setLogLevel, info
from mininet.link import TCLink, Intf
from subprocess import call

def myNetwork():

    net = Mininet( topo=None,
                   build=False,
                   ipBase='10.0.0.0/8')

    info( '*** Adding controller\n' )
    c0=net.addController(name='c0',
                         controller=RemoteController,
                         ip='127.0.0.1',
```

```
                      protocol='tcp',
                      port=6633)

    info( '*** Add switches\n')
    s1 = net.addSwitch('s1', cls=OVSKernelSwitch, dpid='0000000000000001')
    s2 = net.addSwitch('s2', cls=OVSKernelSwitch, dpid='0000000000000002')
    s3 = net.addSwitch('s3', cls=OVSKernelSwitch, dpid='0000000000000003')

    info( '*** Add hosts\n')
    h1 = net.addHost('h1', cls=Host, ip='10.0.0.1', defaultRoute=None)
    h2 = net.addHost('h2', cls=Host, ip='10.0.0.2', defaultRoute=None)
    h3 = net.addHost('h3', cls=Host, ip='10.0.0.3', defaultRoute=None)
    h4 = net.addHost('h4', cls=Host, ip='10.0.0.4', defaultRoute=None)

    info( '*** Add links\n')
    net.addLink(h1, s1)
    net.addLink(s1, s2, bw=20, loss=5)
    net.addLink(s2, h2)
    net.addLink(h4, s1)
    net.addLink(s1, s3, bw=20, loss=5)
    net.addLink(h3, s3)
    net.addLink(s2, s3)
    info( '*** Starting network\n')
    net.build()
    info( '*** Starting controllers\n')
    for controller in net.controllers:
        controller.start()

    info( '*** Starting switches\n')
    net.get('s1').start([c0])
    net.get('s2').start([c0])
    net.get('s3').start([c0])

    info( '*** Post configure switches and hosts\n')

    CLI(net)
    net.stop()

if __name__ == '__main__':
    setLogLevel( 'info' )
    myNetwork()
```

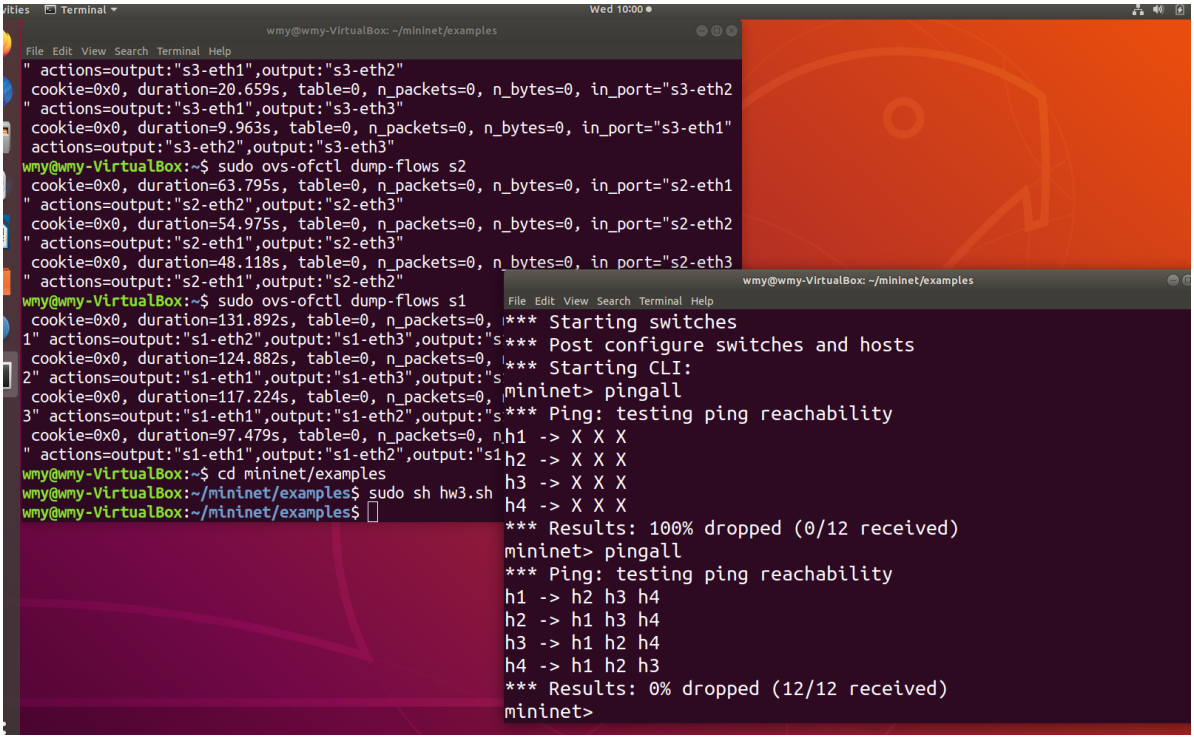
after using "pingall"



hw3.sh

```
#!/bin/sh
sudo ovs-ofctl add-flow s1 in_port=1,actions=2,3,4
sudo ovs-ofctl add-flow s1 in_port=2,actions=1,3,4
sudo ovs-ofctl add-flow s1 in_port=3,actions=2,1,4
sudo ovs-ofctl add-flow s1 in_port=4,actions=2,3,1
sudo ovs-ofctl add-flow s2 in_port=1,actions=2,3
sudo ovs-ofctl add-flow s2 in_port=2,actions=1,3
sudo ovs-ofctl add-flow s2 in_port=3,actions=2,1
sudo ovs-ofctl add-flow s3 in_port=1,actions=2,3
sudo ovs-ofctl add-flow s3 in_port=2,actions=1,3
sudo ovs-ofctl add-flow s3 in_port=3,actions=2,1
```

the outputs



After we add another link between s2 and s3, the hosts' connectivity is broken.

So we use OVS commands to set up flow rules and restore the connectivity.