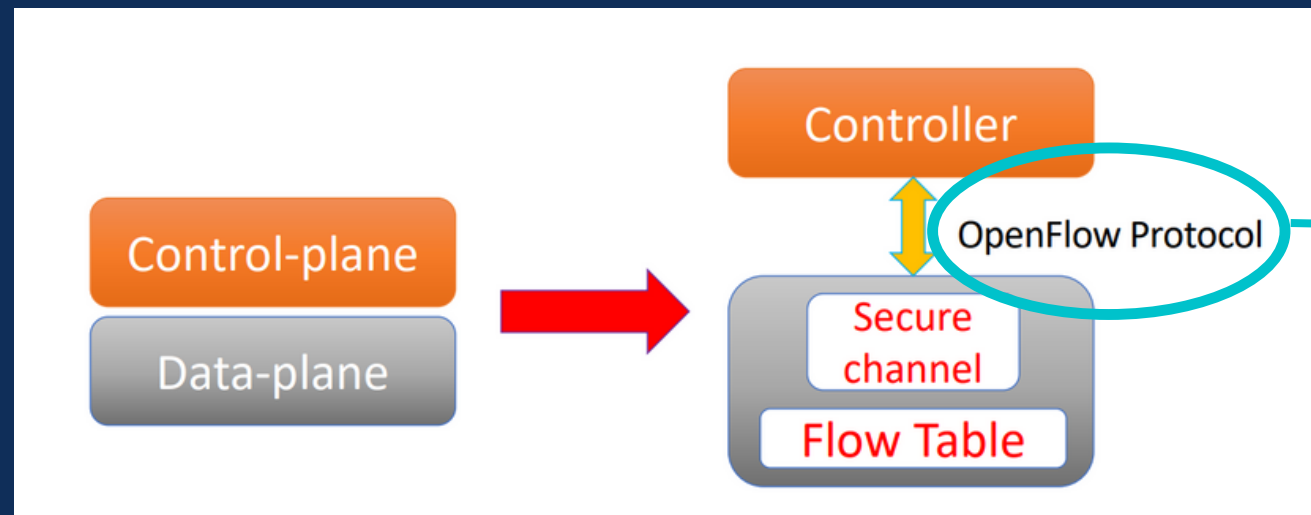


Softwarized and Virtualized Mobile Networks

INTERACTIVE TOPOLOGY DISPLAY FOR SDN NETWORKS

SDN AND REST API

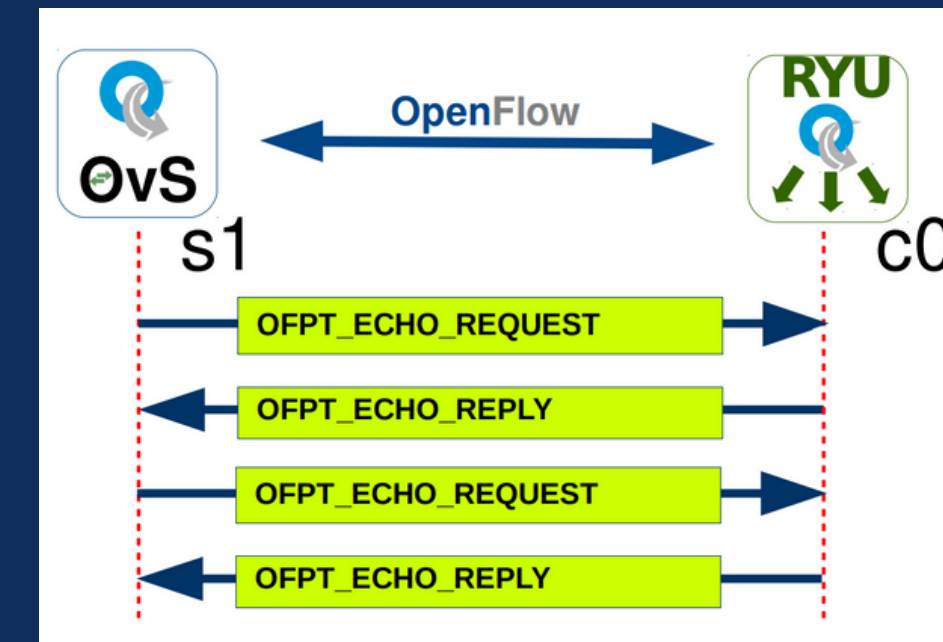


Separation of Control and Data Plane

Send/receive forwarding rules from controller to switches

SDN CONTROLLER -> RYU MANAGER

Ryu manager SDN controller communicate informations with switches and hosts with southband interface



RESTful API-> REQUEST LIBRARY

REST API respond in JSON format from an HTTP GET request

PYTHON FILES CREATED

request.py

JSON response to be
processed by topo_disp.py

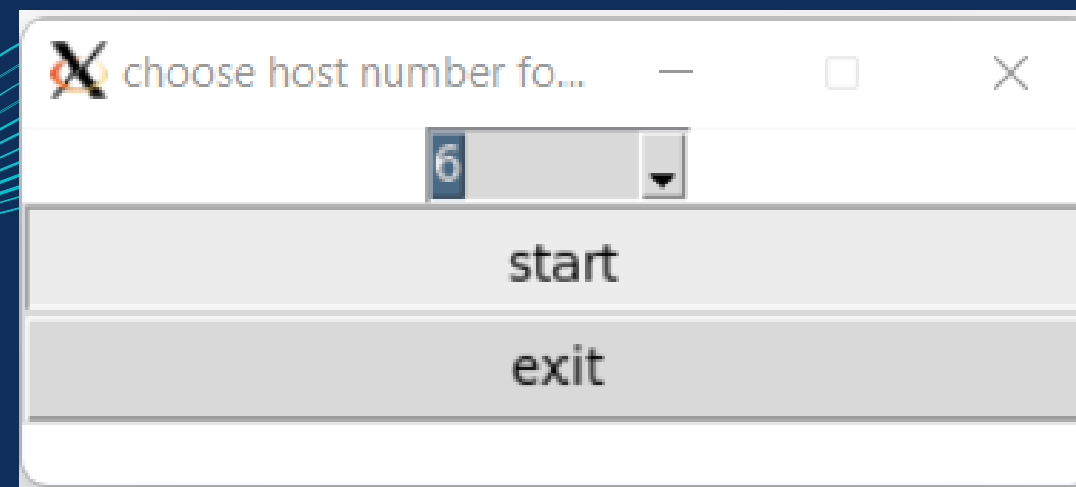
```
import requests

SERVER = 'http://localhost:8080/'

def get_switches():
    r = requests.get(SERVER + 'v1.0/topology/switches')
    if r.status_code == 200:
        return r.json()
    else:
        return None
```

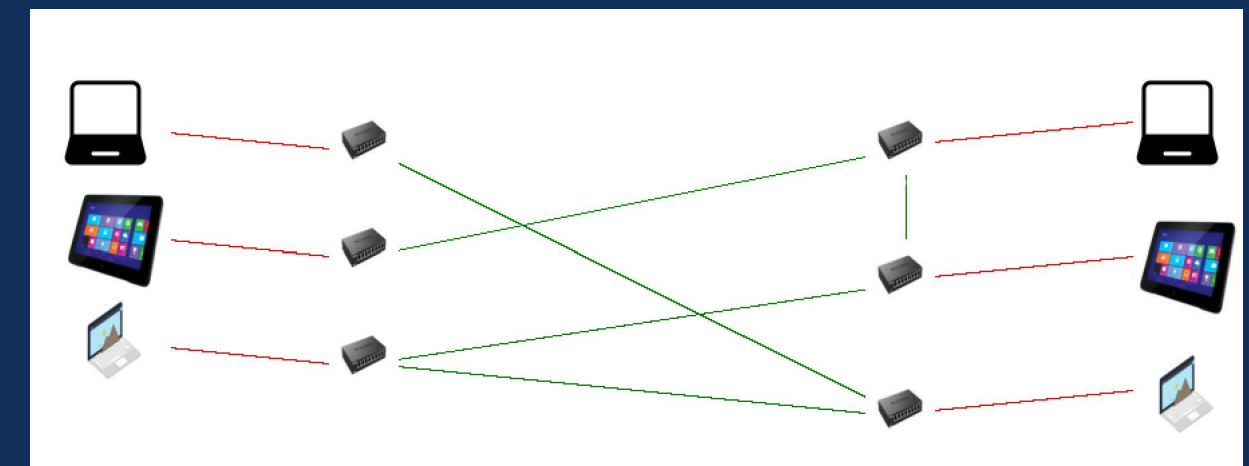
topology.py

Tkinter graphical library
create an interface to
choose host number for
topology



topo_disp.py

Graphical topology viewer
real time update,
capture JSON response



START PROJECT

Run ryu manager

```
ryu-manager gui_topology/gui_topology.py  
simple_switch_13.py --observe-links
```

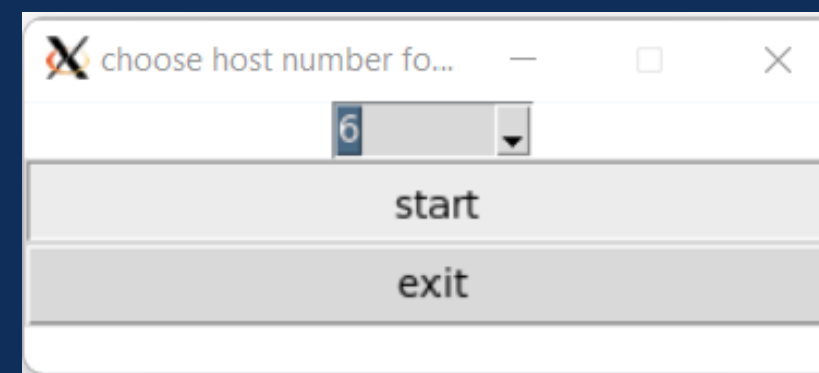
Start RYU controller
with gui topology

Run topology.py

```
sudo python3  
topology.py
```

Start Mininet with a
linear topology of
selected # of hosts

Choose host number for linear topology



Run topo_disp.py

```
sudo python3 topo_disp.py
```

request.py

Return a JSON response

Start graphical
interface to show
topology in real time

JSON RESPONSE OUTPUT

PORTS HW ADDR

PORTS INFO

NAME

FLOW TABLE

} SWITCH INFO

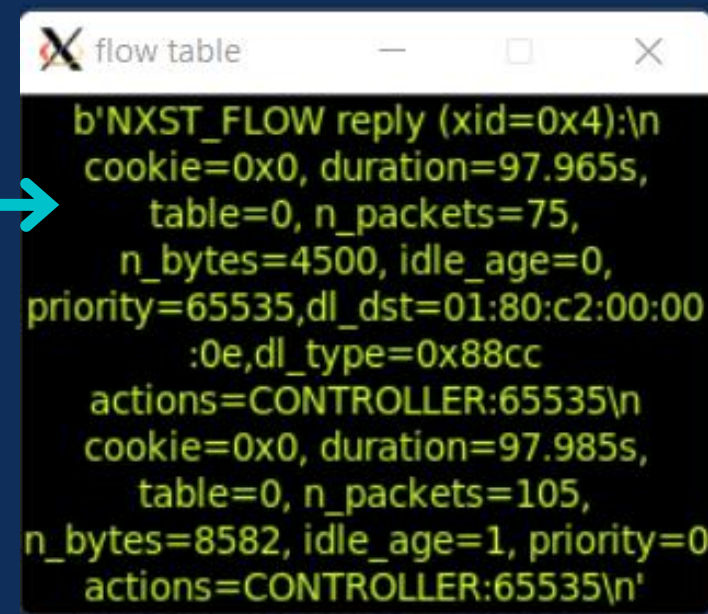
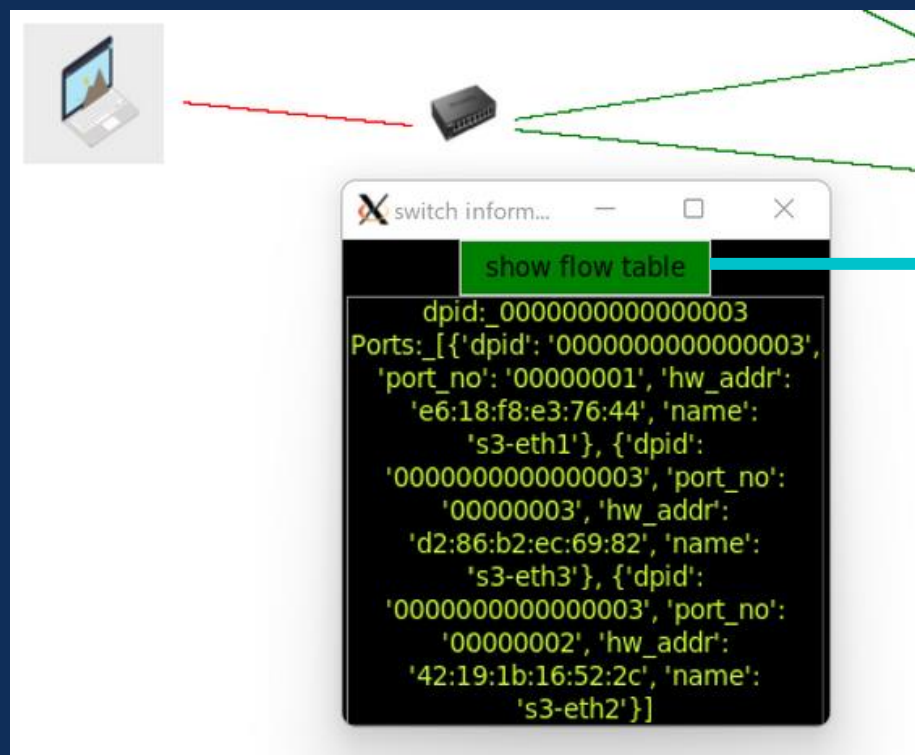
MAC ADDR

IPV4/V6

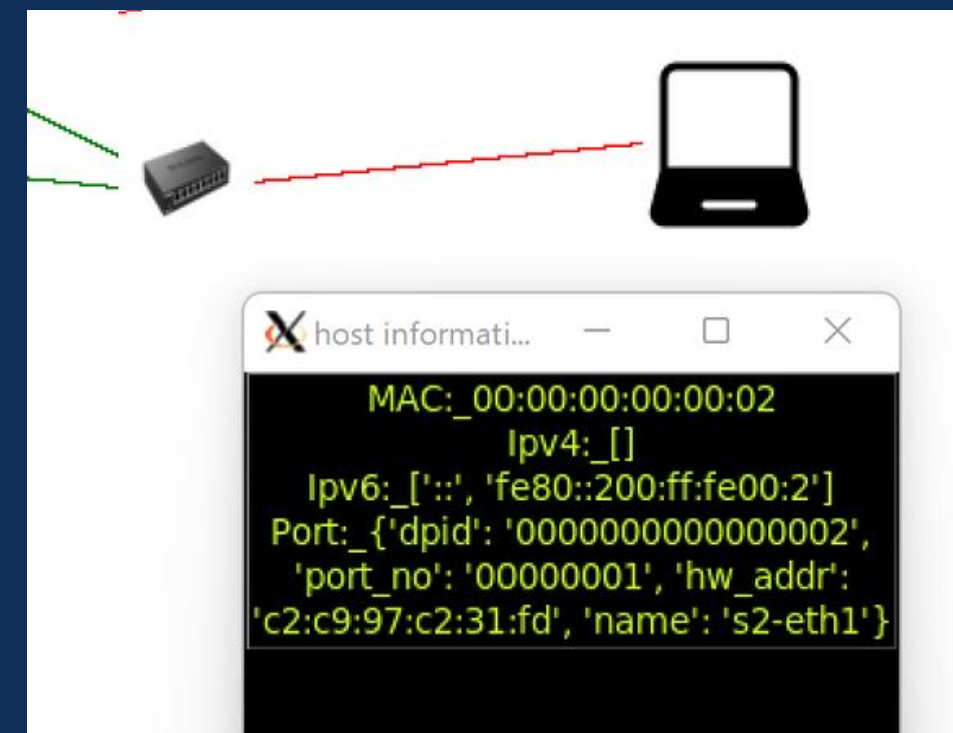
NAME

PORT HW ADDR

} HOST INFO



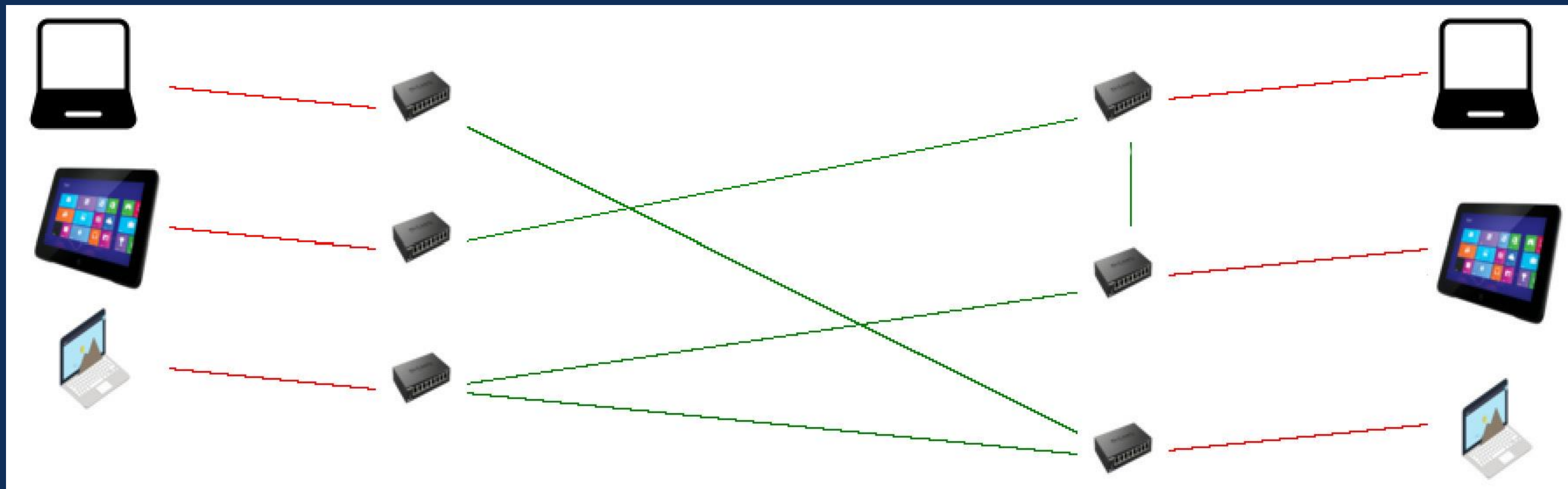
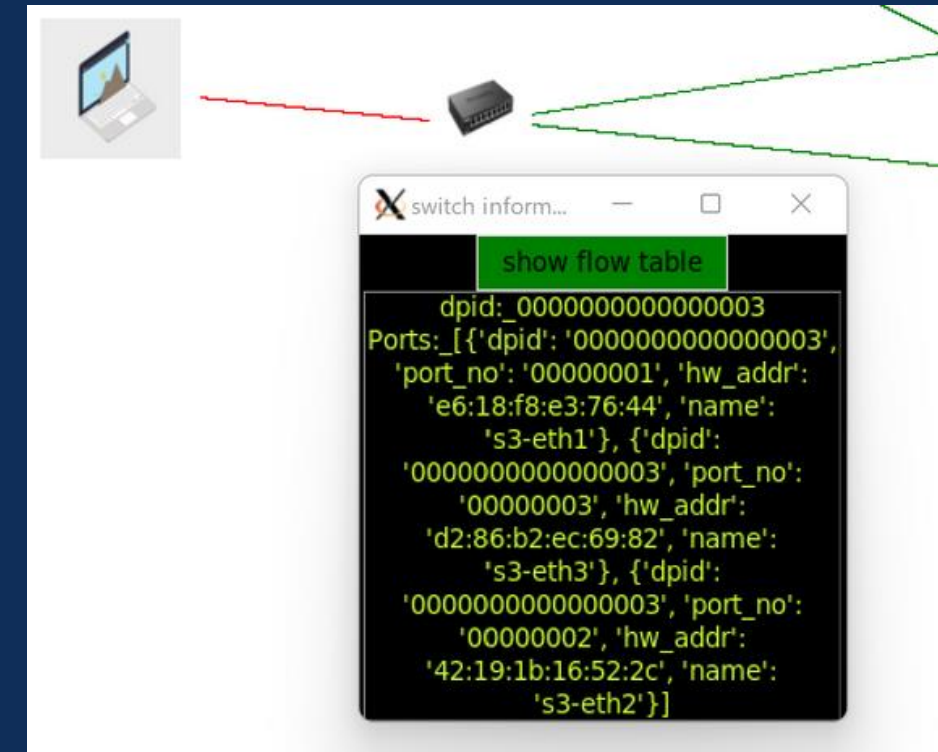
sudo ovs-ofctl dump-flows switch



GRAPHICAL TOPOLOGY

REAL TIME UPDATE

CLICK ON HOSTS OR SWITCHES TO SEE INFORMATIONS



Example: linear topology with 6 hosts