Software Defined Network and simulation using mininet

Aswanth P P
15CO112
CO390 - Seminar

Data Center Networks

Traffic Inside a Data Center

Packet transfer rate within datacenter is very much higher normal rate

Network topology within the organisation wont change often

Networks in data centers need special attention

Data Center Network Opportunities and Challenges

Opportunities

Single Owner	**Easy to bring quick changes**
Central Control	**Centralized management possible**
Well Defined Events	**Optimize networking**

Challenges

Large Scale	**Scalability of Networks**
Takes years to change	**Coupling among specific hardware and software**
Multi-tenancy	**Multi-domain networking**

Networks in Internet of Things

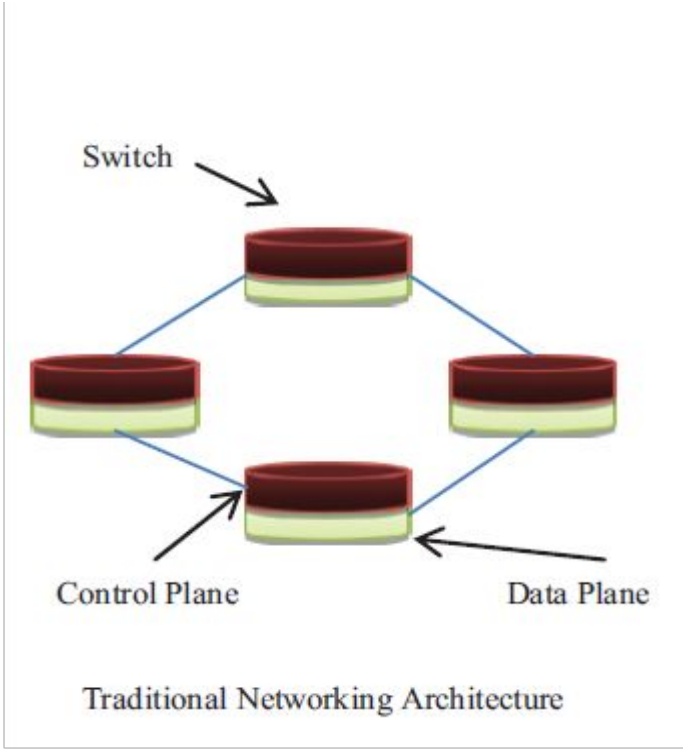
- Enables smart cities, intelligent driving, elder assistance, smart health care, self driven vehicles - depend heavily on the availability of scalable and reliable network.
- Heterogeneous devices and traffic types
- Large scale (50 billion by 2020)
- Dynamic control of network

Features of Future Networks

- Flexible – support different topologies, routing architecture
  Manageable – i.e. separate the policy from the mechanism that implements it
- Scalable – more VM's, IoT devices
- Programmable – not just configure and use standard protocol but program because our network is different from other networks
- Interoperability– support different technologies

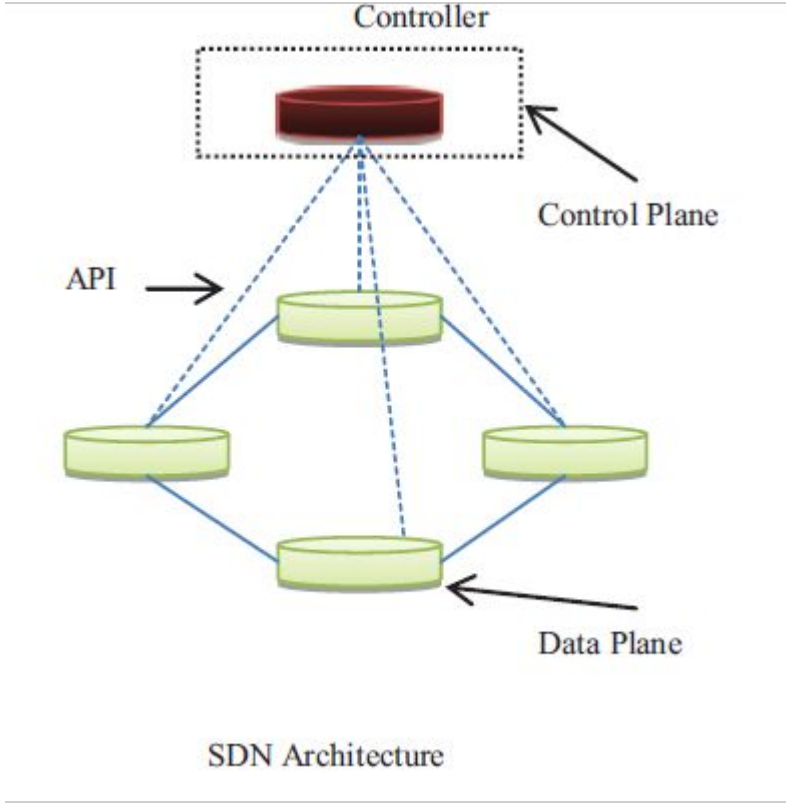
Software Defined Network

Traditional Network



Traditional Networking Architecture

SDN



SDN Architecture

Characteristics of SDN

- Separation of control and data plane
- Centralized control
- Programmable, dynamic changes to network state

Southbound API

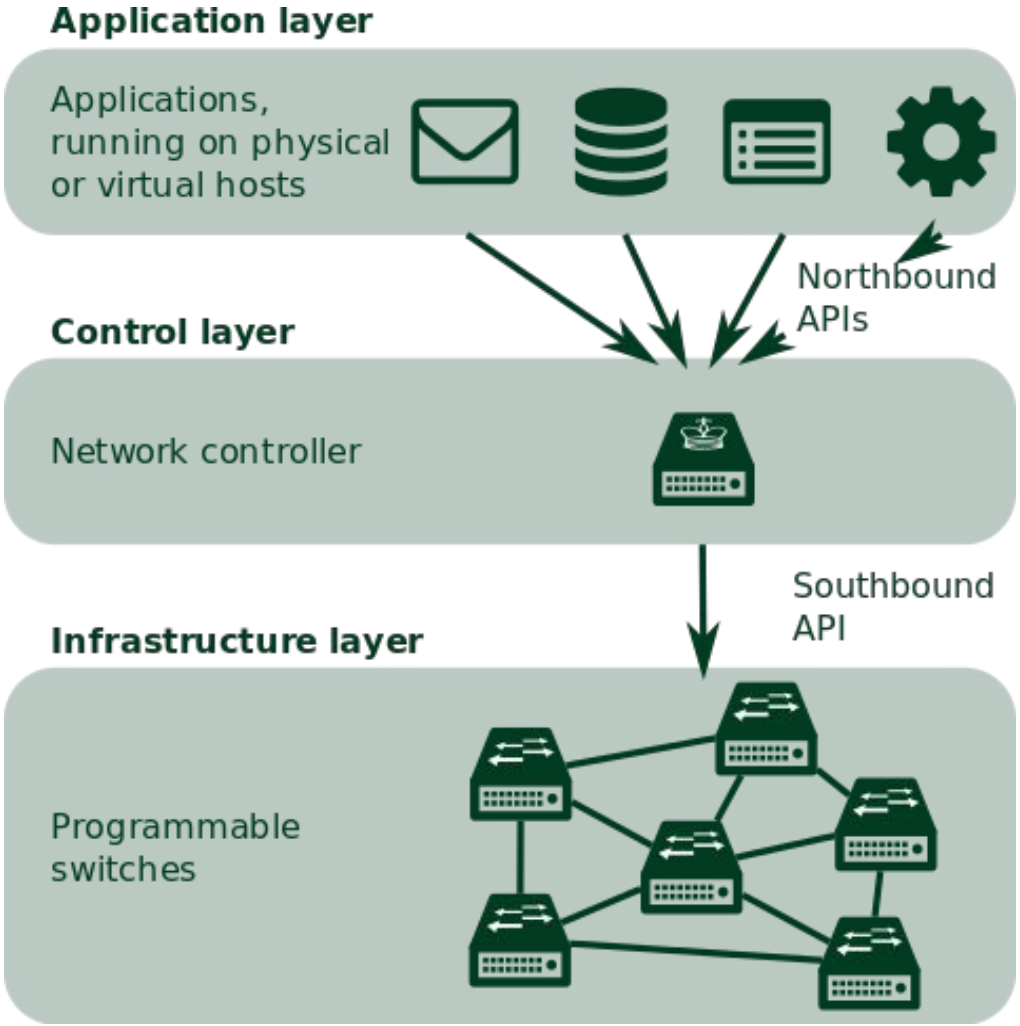
- Controller to network element-OpenFlow

Northbound API

- Controller to application

SDN Sample topology

- Application here may be either firewall,NAT and other network function utilities
- Switches will contact controller only if there is no flow table entry for incoming packet's [O,D] pair
- Controller can be programmed by the applications to give priorities to a flow



SDN packet forwarding rule

1. Forward rule is installed on all switches along the path

2. First packet of flow – contact controller

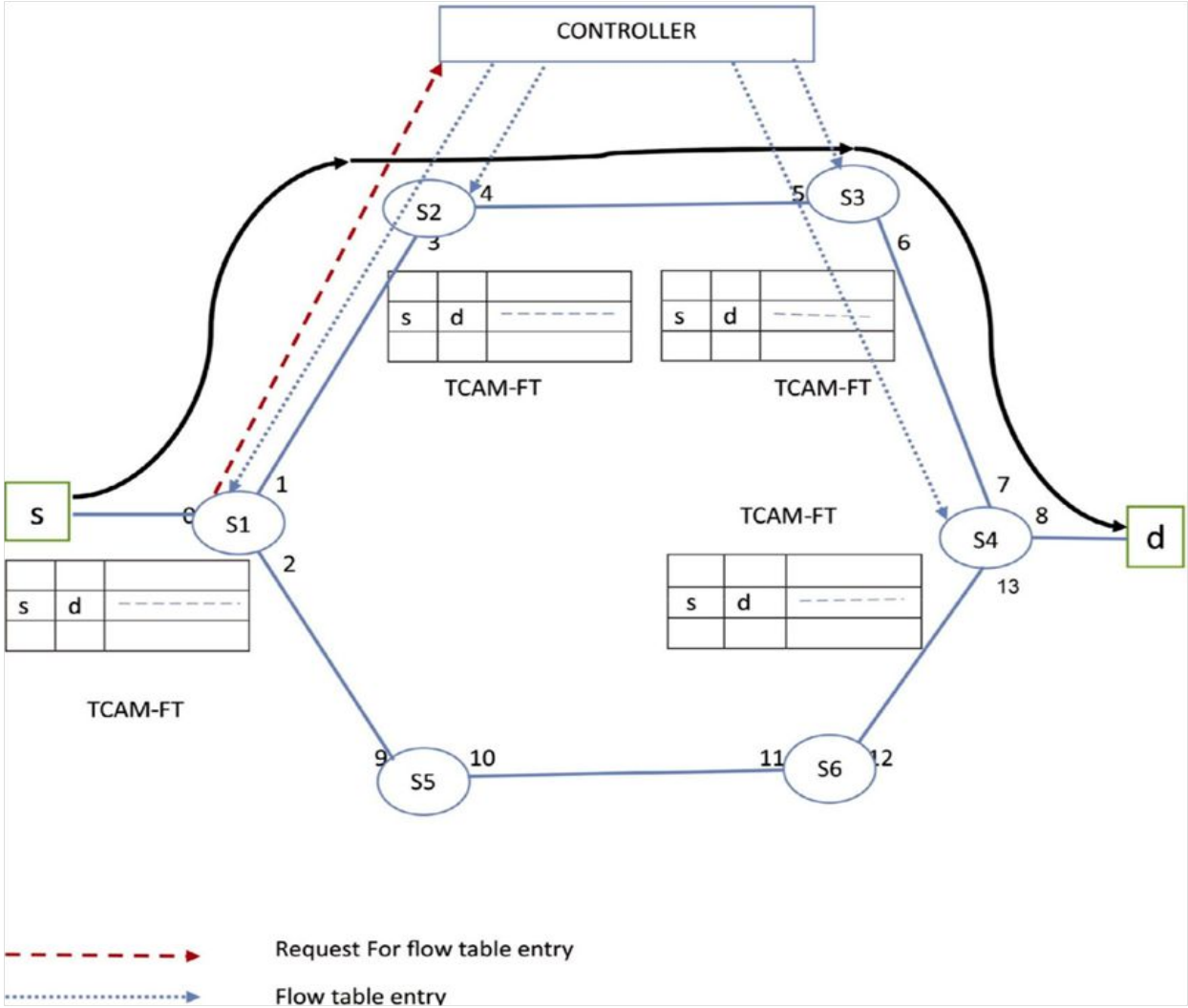
3. Later packets of flow – forwarded by switch without controllers intervention

Routing in SDN

S -> Source Nodes

D-> Destination Node

S1,S2,S3,S4,S5 and S6 are SDN capable switches



Switch Evolution

Black boxes-> Proprietary with very little access (Cisco)

White boxes-> Merchant silicon bare metal switch (Broadcom) with user loaded software (Cumulus OS)

Open vSwitch - SDN support Enabled

Open vSwitch - Switch in SDN

Software switch

Enables communication between

- Co located virtual machines
- Virtual machines on different hosts

Control Plane

Slow / flexible plane

User space

Data Plane

Fast plane

Kernel Space

How to Experiment on SDN without procuring SDN capable hardwares

Mininet Emulator

- It is a network emulator which creates a network of virtual hosts, switches, controllers, and links.
- Mininet hosts run standard Linux network software

Installation

Installing mininet from source files

1. Clone the github repository :
   ```
   git clone git://github.com/mininet/mininet
   ```

2. Change directory to cloned repository :
   ```
   cd mininet
   ./util/install.sh -a
   ```

3. Verify the installation
   ```
   sudo mn --test pingall
   ```

How to use mininet

Launch topology in mininet : sudo mn

A default topology will be loaded which has two hosts that are connected to a switch(OVSSwitch) and a default controller(ovs-controller)

Sample commands available in mininet

1. mininet > nodes : To display all nodes
2. mininet > net : To display all links
3. mininet > dump : To display information about all nodes

Use help to get more CLI commands.

Designing a Custom Topology

Mininet provides some inbuilt topologies like linear, single, minimal, reversed, torus and tree.

Example:

sudo mn --topo linear,4

creates a topology of 4 nodes, each connected with a separate switch

sudo mn --topo single,4

creates a topology 4 nodes, each connected with a single switch

To change the link parameters

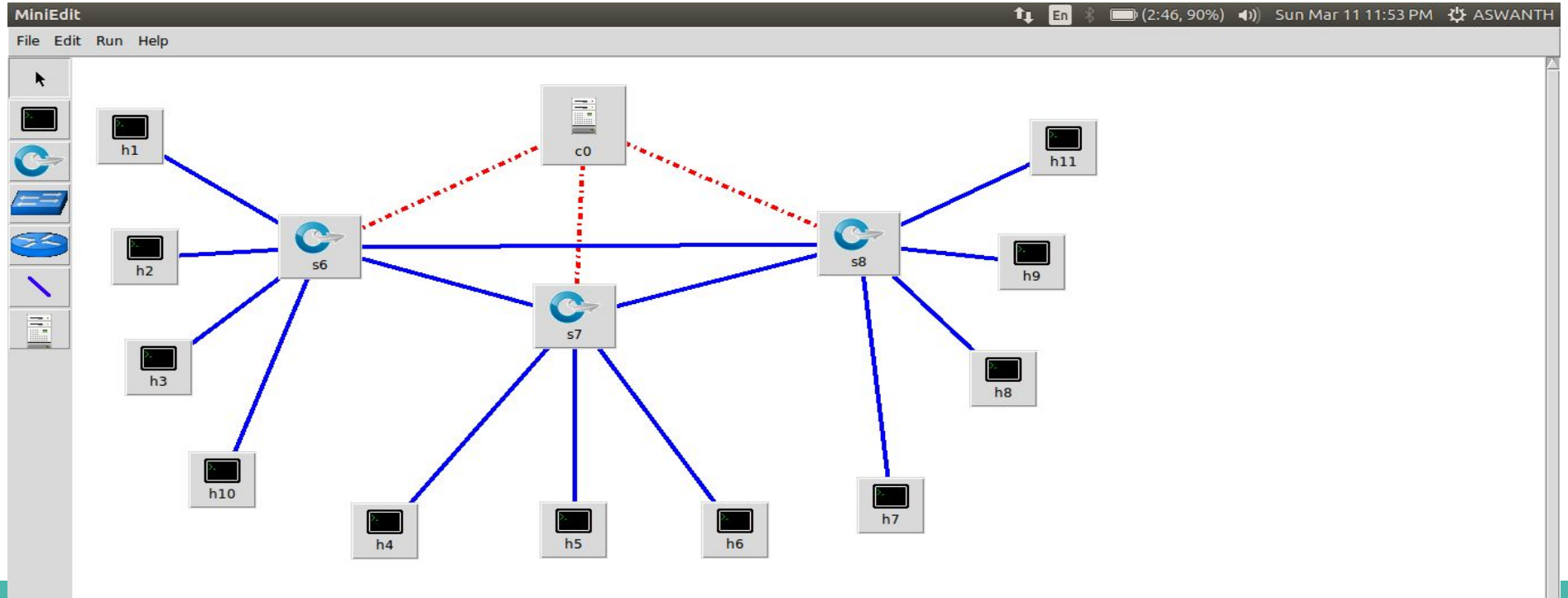
sudo mn --link tc,bw=10,delay=10ms

above command will set bandwidth to 10Mbps and delay to 10ms for all the links in network

Creating Topology using mininet GUI tool

Redirect your terminal to mininet directory and run this command

sudo python ./examples/miniedit.py



Launch custom topology in mininet

1. **Topology created by GUI tool** can exported to python script by selecting File-> Export Level 2 Script option from the above GUI and save it in custom folder of mininet
2. **Two ways to launch the topology**
   a. sudo mn --custom ./custom/topo-2sw-2host.py --topo mytopo

   Here topology name is mytopo and saved in custom folder

   b. sudo python ./custom/mycustom.py

   This command is useful only if mycustom.py is exported as level 2 script from GUI tool

Installing Different Controllers

Mininet supports different type of controller POX ,NOX etc

Installing POX controller

```
git clone http://github.com/noxrepo/pox
```

```
cd pox
```

```
sudo ./pox.py forwarding.l2_learning
```

Above command will download and start pox controller

Execution of mininet with some examples

Now, I would like to demonstrate the following:

1. How to create a topology
2. How to load topology into mininet
3. How to generate traffic between nodes
4. How to capture packets in a node
5. How to test a network link

References

- Why mininet?: http://openvswitch.org/support/ovscon2015/16/1305-lantz.pdf
- Mininet installation: http://mininet.org/download/
- Mininet commands (topology, link variation, etc): http://mininet.org/walkthrough/
- Miniedit: http://www.brianlinkletter.com/how-to-use-miniedit-mininets-graphical-user-interface/
- Mininet with POX controller: http://www.brianlinkletter.com/using-the-pox-sdn-controller/

THANK YOU

ASWANTH P P (15CO112)
B.Tech ,Dept of CSE ,NITK Surathkal
ppaswanth3@gmail.com