

NWEN 302

Lab 4

Wael Aldroubi  
300456658

What is SDN? Why to use SDN? How it is different from traditional network topology nowadays?

Nowadays networks depends on hardware and software to direct the traffic through a series of routers and switches, using control plane to decide the best route and to manage the logic of how data will flow through the network like OSPF, while data plane will manage the forwarding tables like MAC tables.

Both of control plane and data plane are always connected and depending on each other in one single node, every node has its own policies that will make the network complicated and harder to implement.

Control plane

Control plane

Control plane

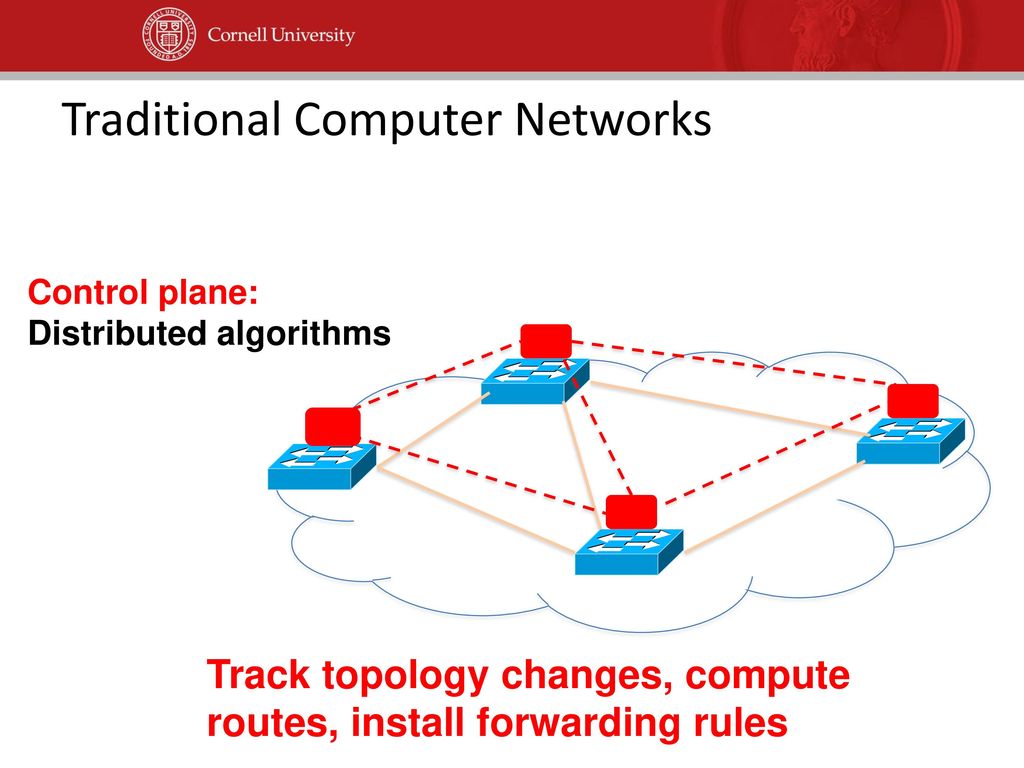
Control plane

Data plane

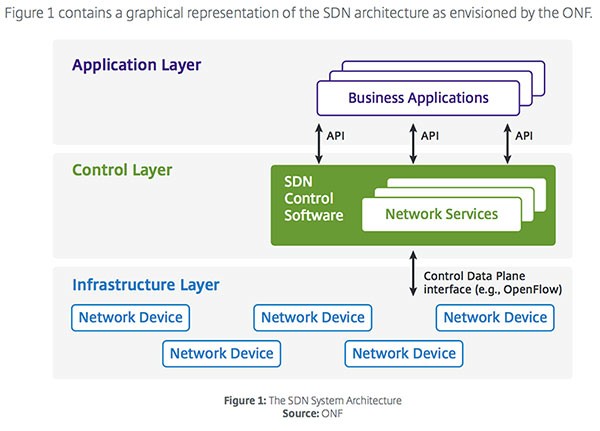
Data plane

Data plane

Data plane

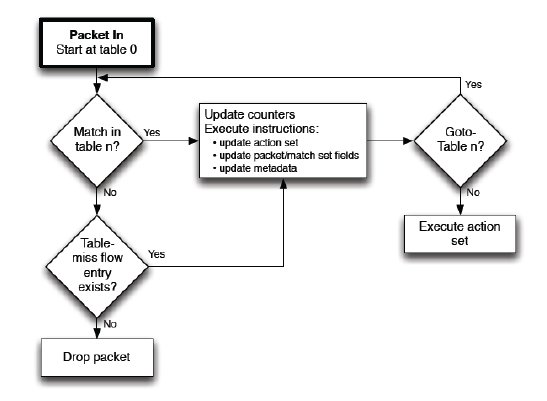


While SDN is similar to OS (operation system).

1. We have the on the top network applications.
2. Then we have application interfaces to manage the connection between applications and network controller (network operating system) like java API or REST.
3. The SDN controller: control the logic of data flow, has services like  
   a. topology services to monitor and discover the network.  
   b. inventory services to track forwarding devices and store information about them.  
   c. statistics services to control traffic counters.  
   d. host tracking to keep track of IP and MAC addresses located on the network.
4. Then we have southbound interface: to control the connection between the controller and forwarding devices, like Open Flow, OVSDB, NETCONF and SNMP.
5. Forwarding devices: to control data plane like switches.

How SDN works?

Data will arrive at the forwarding devices, if the destination address is in the forwarding table it will send it straight away to destination.

If the data has no entry it will send it to the controller to define what is the source, then It will be send back to the controller to forward it again, then It does not require to send it again to the controller, in the forwarding device the packet will either forwarded or dropped or sent to the controller.

At the first the switch will connects to the controller, then all unknown packets will be sent to the controller, the controller will find routes for unknown packets and unknown packets will become known, once the controller finds the router and installs the rule for the unknown flow, the flow will not be sent to the controller.

Explanation of the code (RYU, Open Flow):

In simple switch we defined the host’s ports connected to, every mac of a device is connected to a port.

Add flow will check new packets to the forwarding table.

Packet handler is to control the flow of the data between host and forwarding table.

……………………………….

For task 1:

Created two functions, block packets and block flow, both are called to stop data flow between the two devices.

……………………………….

For task 2:

In the packet status handler, a request defined to show the traffic going to and originating from host 1.

……………………………….

For task 3:

Port status replay handler, get topology data and LLDP packet will handle Keeping track of all traffic, Topology discovery and LLDP to discover connected devices and where data is arriving.

……………………………….

Code source:

(1) <https://github.com/osrg/ryu/blob/master/ryu/app/simple_switch_stp.py>

(2) <https://osrg.github.io/ryu-book/en/html/traffic_monitor.html>

(3) <http://sdn-lab.com/2014/12/31/topology-discovery-with-ryu/>

(4) <http://sdn-lab.com/2014/12/25/shortest-path-forwarding-with-openflow-on-ryu/>

Resources:

(1) <https://www.youtube.com/watch?v=DiChnu_PAzA>  
(Introduction to SDN (Software-defined Networking)

(2) <https://www.youtube.com/watch?v=Z5Gi2Bpd82M>  
(What is software-defined networking (SDN)?)

……………………………….

# References

*http://sdn-lab.com/2014/12/25/shortest-path-forwarding-with-openflow-on-ryu/*. (n.d.). Retrieved from http://sdn-lab.com/2014/12/25/shortest-path-forwarding-with-openflow-on-ryu/

*http://sdn-lab.com/2014/12/31/topology-discovery-with-ryu/*. (n.d.). Retrieved from http://sdn-lab.com/2014/12/31/topology-discovery-with-ryu/

*https://github.com/osrg/ryu/blob/master/ryu/app/simple\_switch\_stp.py*. (n.d.). Retrieved from https://github.com/osrg/ryu/blob/master/ryu/app/simple\_switch\_stp.py

*https://osrg.github.io/ryu-book/en/html/traffic\_monitor.html*. (n.d.). Retrieved from https://osrg.github.io/ryu-book/en/html/traffic\_monitor.html

*https://www.youtube.com/watch?v=DiChnu\_PAzA*. (n.d.). Retrieved from https://www.youtube.com/watch?v=DiChnu\_PAzA

*https://www.youtube.com/watch?v=Z5Gi2Bpd82M*. (n.d.). Retrieved from https://www.youtube.com/watch?v=Z5Gi2Bpd82M