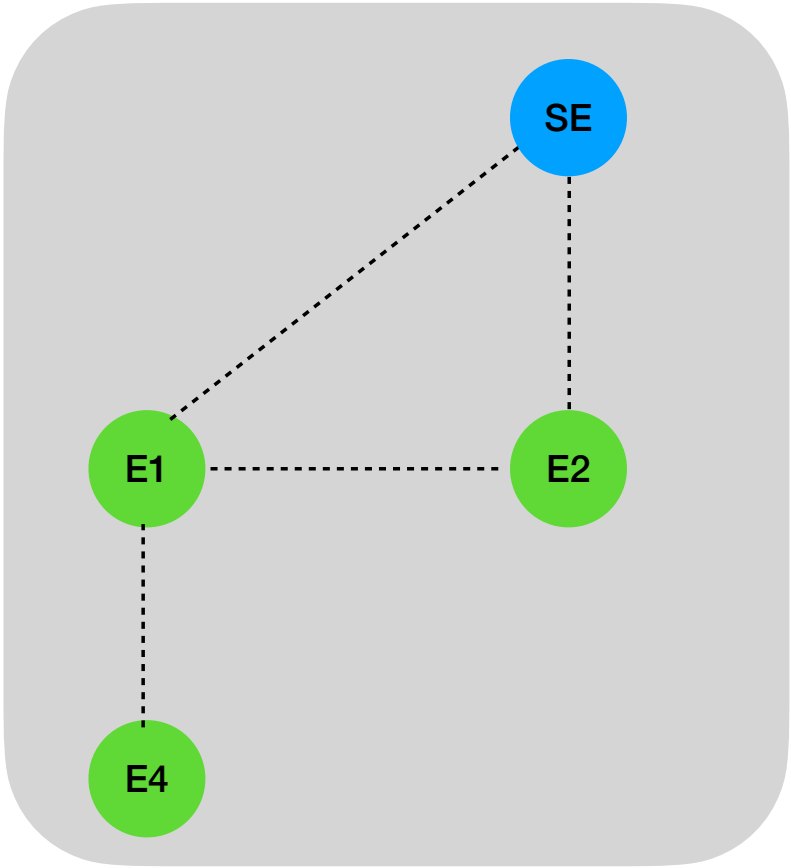
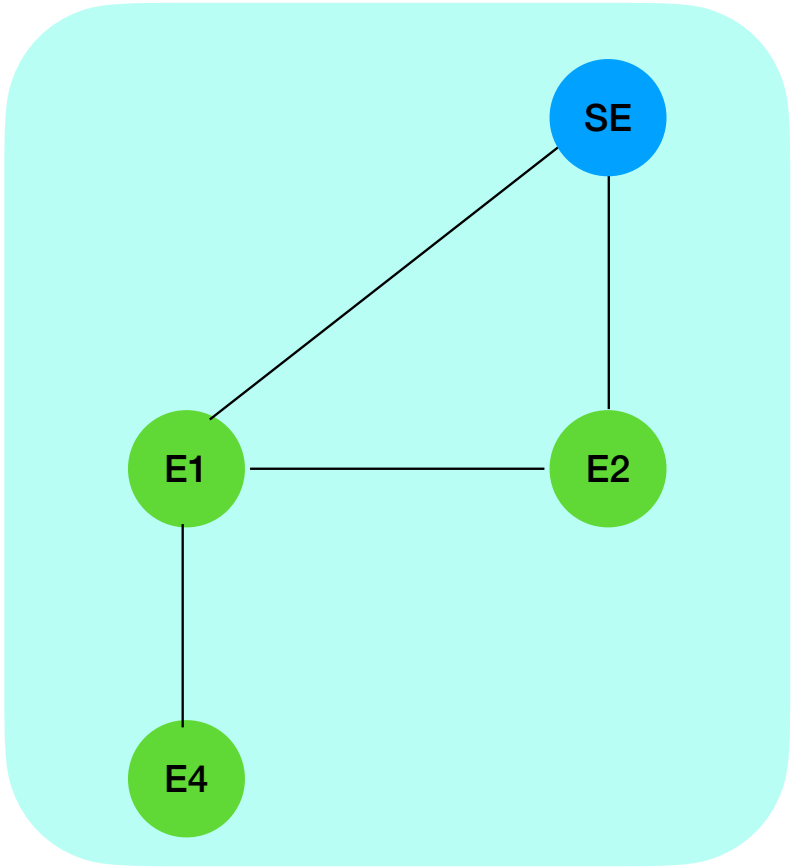
Plan-C

-How to setup the SADEdge testbed (Toy topology)

Network Topology



Control Plane



Data Plane

MAC address

Control Plane Interfaces

E1 MAC address : ”98:48:27:e1:32:ad"

E4 MAC address : “98:48:27:e2:c2:0f"

E2 MAC address : “98:48:27:e2:c6:5b"

SE MAC address: ”00:0f:00:14:91:d7”

MAC address

Data Plane Interfaces

E1 MAC address : "98:48:27:e2:bf:82"

E4 MAC address : "98:48:27:e1:39:b9"

E2 MAC address : "98:48:27:e2:e7:dc"

SE MAC address: "00:0f:00:10:10:23"

IP address

Control Plane Interfaces: “wlan2”

E1 IP address : "10.0.0.1"

E2 IP address : "10.0.0.2"

E4 IP address : “10.0.0.4"

SE IP address: "10.0.0.10"

IP address

Data Plane Interfaces : “wlan1”

E1 IP address : "192.168.2.1"

E4 IP address : "192.168.2.4"

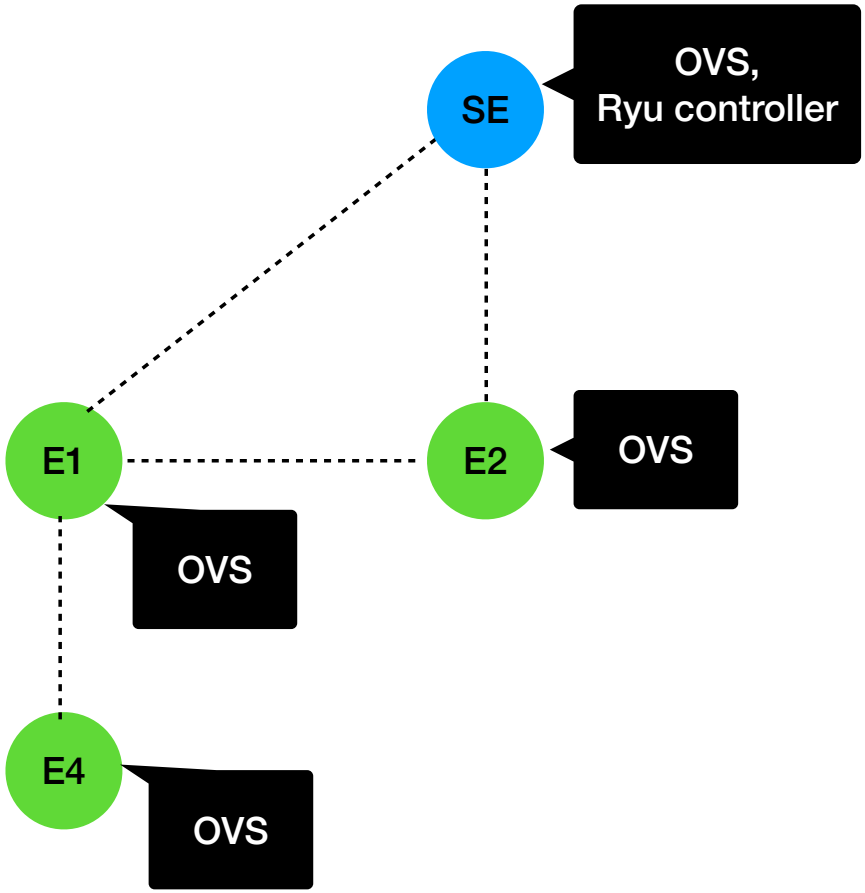
E2 IP address : "192.168.2.2"

SE IP address: "192.168.2.10"

superedge_control_interface="wlx000f001491d7"

superedge_data_interface="wlx000f00101023"

SDN concept

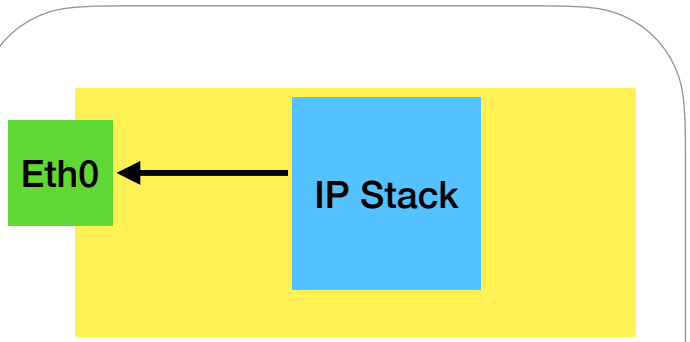


OVS : Open Virtual Switch

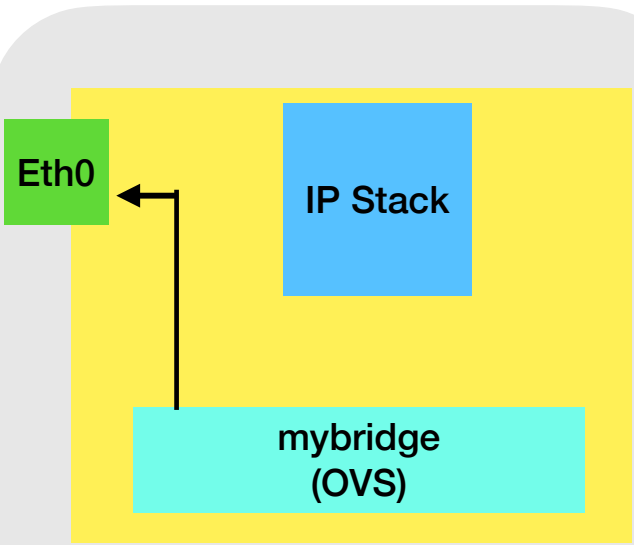
Ryu controller : SDN controller

SDN concept

Create OVS / bridge



E1 (without OVS enable)



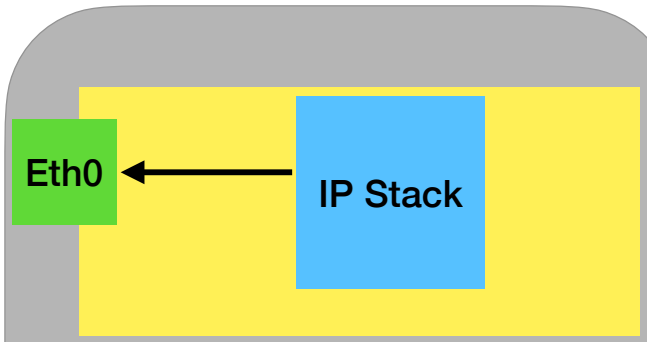
Create OVS mybridge

$ ovs-vsctl add-br mybridge

$ ovs-vsctl show

$ ifconfig mybridge up

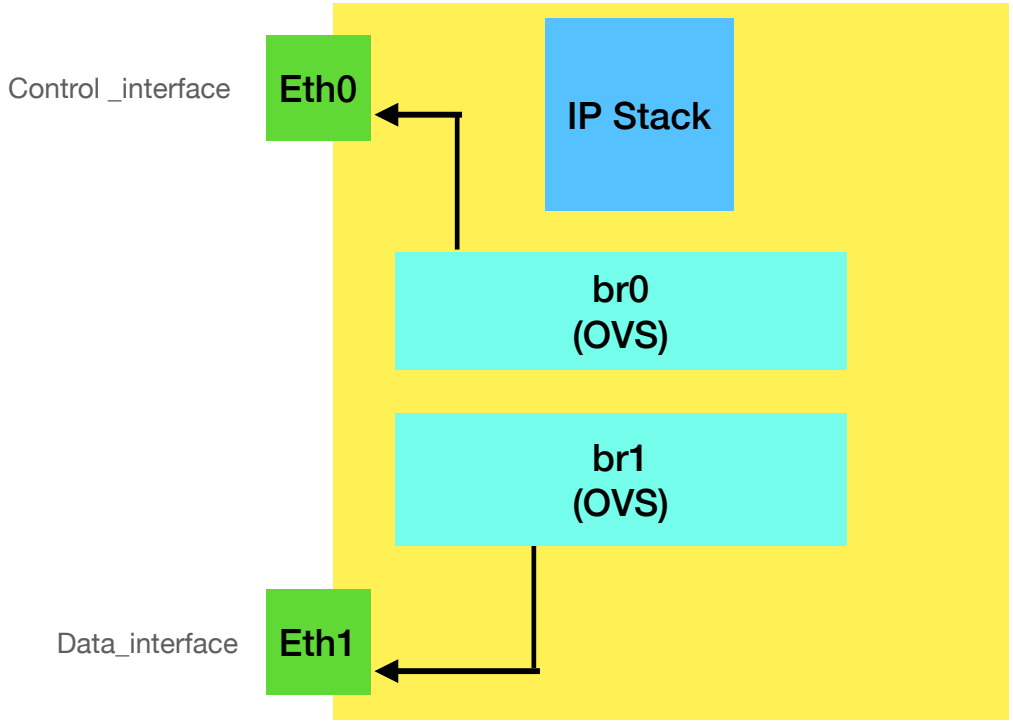
$ ovs-vsctl add-port mybridge eth0



E1 (without OVS enable)

$ ovs-vsctl del-br mybridge

OVS in Edge nodes and SE



Instruction how to set the testbed

- Initial setup the experiments

1> Set edge1,edge2, edge4 and superedge in ad-hoc mode.

2> Do ping tests to check connectivities in ad-hoc network. (In this stage, all ping tests should be successful.) Since all edges are close to each other in indoor testing, edge4 will reach to superedge with one hop. But in the real environment, edge4 cannot reach to superedge with one hop. The primary route for edge4 to superedge is edge4->edge1->superedge.

3> To test primary route, Run edge1_pretest.sh in edge1. Run edge2_pretest.sh in edge2. Run edge4_pretest.sh in edge4. Run se_pretest.sh in superedge. Ping edge4 to superedge, Ping test will be successful. But, if you check with tcpdump or wireshark, you will see that the packets from edge4 goes to edge1 first and edge1 relays that packet to superedge. The Openflow rules written in script files do that primary route.

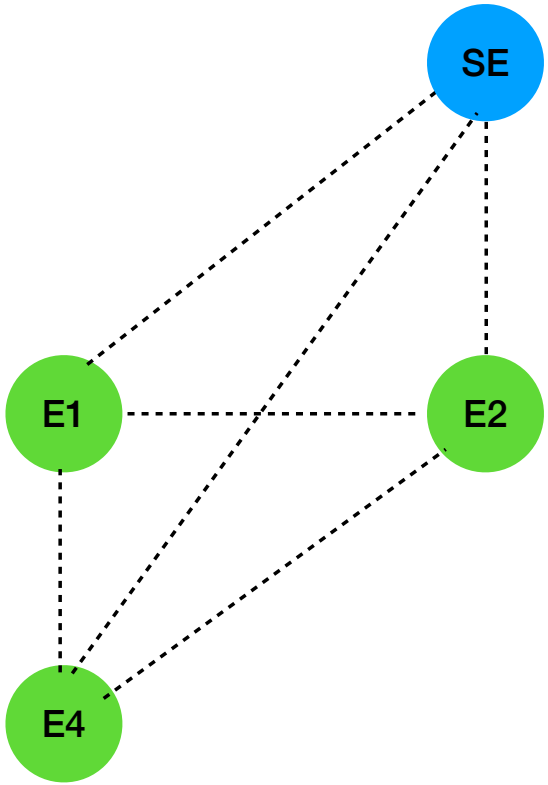
4> Now, the primary routes are established.Then, so that the ovs in the edges connect to the Ryu controller running in superedge, run monitor_test.py in superedge by using the command "ryu-manager monitor_test.py".

5> Then, type the command "sudo ovs-vsctl show" in edges and superedge to check whether the ovs in the edges and superedge are connected to Ryu controller or not. If the result of "sudo ovs-vsctl show" replies the status " is connected : true" , then, all the ovs are connected to Ryu controller.

6> After running monitor_test.py, it will give the datapath-ids of the ovs-bridges and the Openflow port-stats-replies. Copy those datapath-ids and paste it in the superedge.py in the flowrules folder.

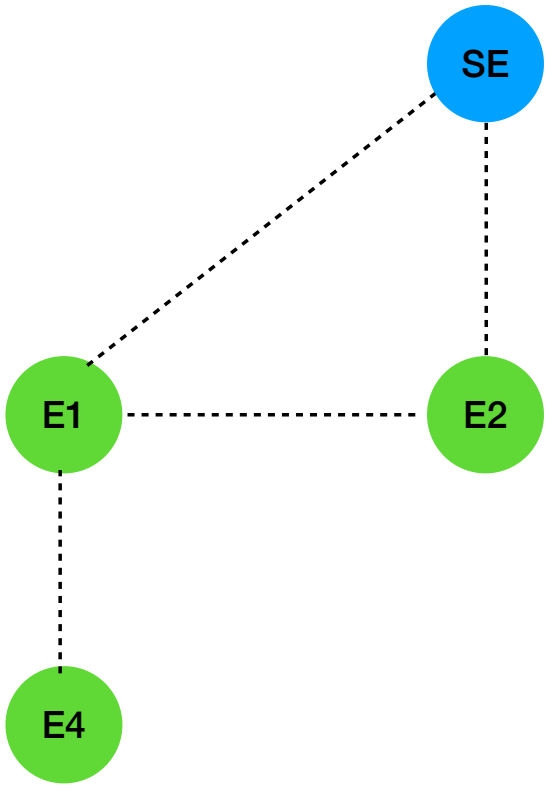
7> Do the above steps for edge3,edge5 and edge6 also.

Ping test (without SDN enable)



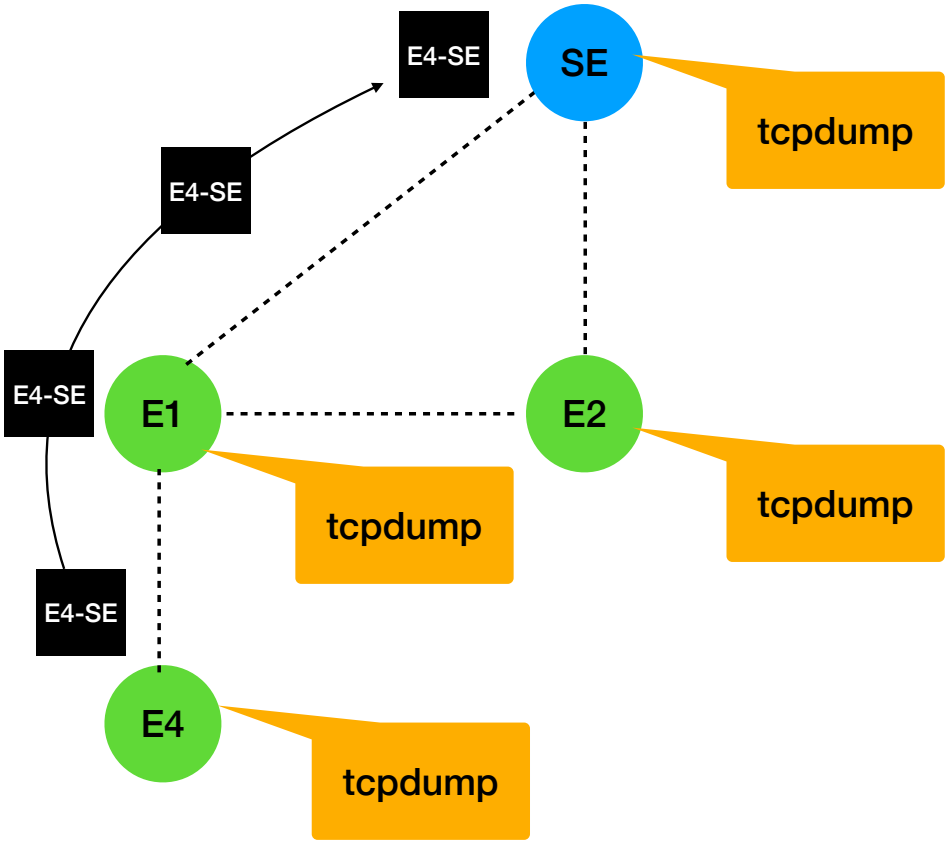
1. Add the IP addresses to all interfaces manually in `/etc/network/interfaces/`
2. Enable ad-hoc mode :
    1. “wlan1” for data plane interfaces
    2. “wlan2” for control plane interfaces
3. Do “ping test” for all possible pair of interfaces
    - Test all control plane interfaces
    - Test all data plane interfaces
4. The results of the ping test will show the successful connectivities of all interfaces

Enable SDN and OpenFlow rules



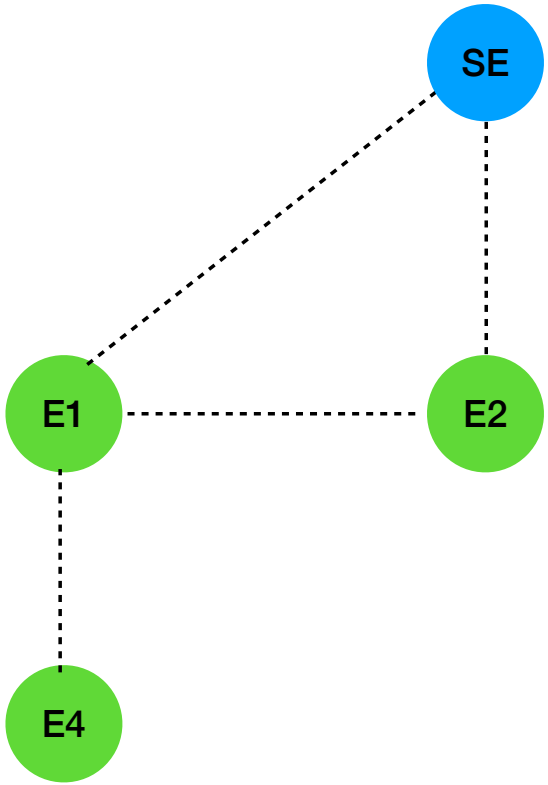
1. Copy shell script files:
   - "edge1_pretest.sh" to E1
   - "edge2_pretest.sh" to E2
   - "edge4_pretest.sh" to E4
   - "se_pretest.sh" to SE
2. Set the permission of the execution on the shell script files in all nodes by command:
   - *sudo chmod +x edge1_pretest.sh*
   - *sudo chmod +x edge2_pretest.sh*
   - *sudo chmod +x edge4_pretest.sh*
   - *sudo chmod +x se1_pretest.sh*
3. Execute all shell script file by command
   - sudo ./edge1_pretest.sh
   - sudo ./edge2_pretest.sh
   - sudo ./edge4_pretest.sh
   - sudo ./se_pretest.sh
4. Do "ping test" for all possible pair of interfaces
   - Test all control plane interfaces
   - Test all data plane interfaces
5. The results of the ping test will show the successful connectivities of all interfaces

Check the correctness of the primary routes



1. Ping test on E4 to SE
2. Do tcpdump or WireShark on E4 to see that the packets is passing node E1 not going to SE directly.
3. Do tcpdump and do the ping test for all edge nodes to SE.

Connect to Ryu Controller



1. OVS in E1, E2 and E4 connect to Ryu controller in SE
2. Run file *"monitor_test.py"* in SE by command:
   - `ryu-manager monitor_test.py`
3. Check whether the OVS in all edge nodes and SE connect to Ryu controller or not by command:
   - `sudo ovs-vsctl show`
4. If the results of "sudo ovs-vsctl show" reply with the status "is connected : true", then all OVS in edge nodes and SE connect to Ryu controller.
5. After running "ryu-manager monitor_test.py", the results will return the "datapath-id" of the ovs-bridges and the OpenFlow port-status-replies.
6. Copy the "datapath-id" of all ova-bridges and paste them in "superedge.py" (in the flowrules directory) line xx to line xx