

## SDN Exercise No. 3

Surname: Yu  
First name: Chunyuan  
ID#: 2587628

Team member: Letian Feng(2255840), Zhen Chen(2665935), Chunyuan Yu(2587628)

### Problem 3.1 - SDN Relatives to OpenFlow

a)

Active Networks, 4D, Signaling System No.7

b) I)

CE:

1. Implement ForCES protocol;
2. Use the implemented ForCES protocol to instruct FE(s) on how to process packets;
3. Execute control and signaling protocols.

FEs:

1. Implement ForCES protocol;
2. Use underlying hardware to provide per-packet processing and handling as directed by a CE via the ForCES Protocol.

II)

1. ForCES aims to define a framework and associated protocol(s) to standardize information exchange between the control and forwarding plane.
2. Separate components would allow component vendors to specialize in one component without having to become experts in all components;
3. Standard protocol also allows the CEs and FEs from different component vendors to interoperate with each other and hence it becomes possible for system vendors to integrate together the CEs and FEs from different component suppliers. This interoperability translates into increased design choices and flexibility for the system vendors.
4. ForCES will enable rapid innovation in both the control and forwarding planes while maintaining interoperability. Scalability is also easily provided by this architecture in that additional forwarding or control capacity can be added to existing network elements without the need for forklift upgrades.

### Problem 3.2 - The OpenFlow Protocol

a)

With the feature "TCP flag matching", an open-flow switch could know when a TCP connection is set up or broken, and then transfer this packet to the controller, so that the controller could modify flow tables of switches on the path from source to end of this packet in order to improve network performance. For example, if a packet matched TCP flag "SYN", the controller increase the priority of the entry with the same destination ip address as in this SYN-packet.

b)

1. Start ingress processing, match packet in ingress table n(n is 0 at first):
  - if matched, then GOTO 2;
  - if not matched, it's a table miss, GOTO 4;
2. Do instructions in the entry, such as:
  - a) Direct packet to another table, and then GOTO 1;
  - b) Stop packet at the table, and then GOTO 3;
3. Stop packet at the table and execute action set, then GOTO 5;
4. Check the table configuration how to handle table miss:

- a) Drop it, end;
  - b) Pass it to another table, GOTO 1 or 6;
  - c) Send it to the controller(s), GOTO 6;
5. Check if there is egress table(s):  
 if no, then forward packet out of port(s) in the action set, end;  
 if yes, GOTO 6;
6. Start egress processing, analogous to ingress processing.

### Problem 3.3 - Flow Spaces

a)

