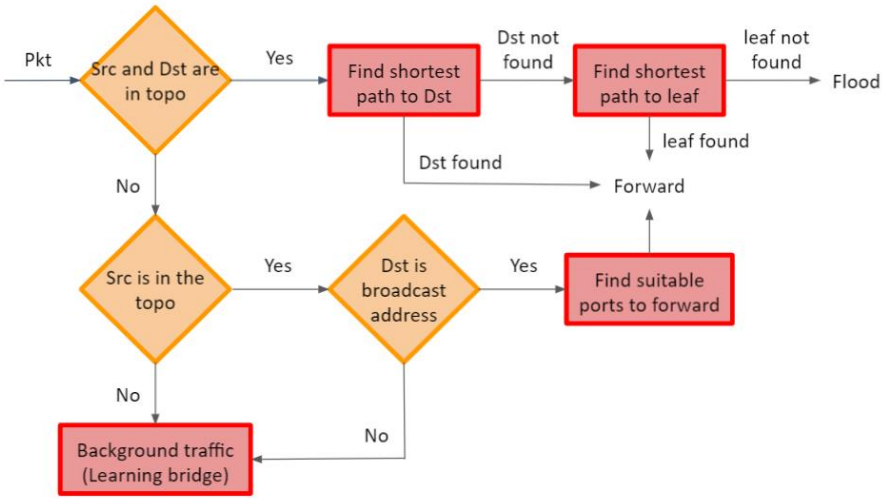
Data Center Network Project 4 Report

309552007

袁鈺勛

A. Explanation of design implementation

1. Flowchart



2. Screenshot

a. Pingall

```
default,protocols=OpenFlow13 --mac --arp
[sudo] password for pncourse:
*** Creating network
*** Adding controller
Connecting to remote controller at 127.0.0.1:6653
*** Adding hosts:
h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11 h12 h13 h14 h15 h16
*** Adding switches:
s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15
*** Adding links:
(s1, s2) (s1, s9) (s2, s3) (s2, s6) (s3, s4) (s3, s5) (s4, h1) (s4, h2)
 (s5, h3) (s5, h4) (s6, s7) (s6, s8) (s7, h5) (s7, h6) (s8, h7) (s8, h8
) (s9, s10) (s9, s13) (s10, s11) (s10, s12) (s11, h9) (s11, h10) (s12,
h11) (s12, h12) (s13, s14) (s13, s15) (s14, h13) (s14, h14) (s15, h15)
(s15, h16)
*** Configuring hosts
h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11 h12 h13 h14 h15 h16
*** Starting controller
c0
*** Starting 15 switches
s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15 ...
*** Starting CLI:
mininet> pingall
*** Ping: testing ping reachability
h1 -> X X h4 X X h7 X X h10 X X h13 X X h16
h2 -> X X X h5 X X h8 X X h11 X X h14 X X
h3 -> X X X X h6 X X h9 X X h12 X X h15 X
h4 -> h1 X X X X h7 X X h10 X X h13 X X h16
h5 -> X h2 X X X X h8 X X h11 X X h14 X X
h6 -> X X h3 X X X X h9 X X h12 X X h15 X
h7 -> h1 X X h4 X X X X h10 X X h13 X X h16
h8 -> X h2 X X h5 X X X X h11 X X h14 X X
h9 -> X X h3 X X h6 X X X X h12 X X h15 X
h10 -> h1 X X h4 X X h7 X X X X h13 X X h16
h11 -> X h2 X X h5 X X h8 X X X X h14 X X
h12 -> X X h3 X X h6 X X h9 X X X X h15 X
h13 -> h1 X X h4 X X h7 X X h10 X X X X h16
h14 -> X h2 X X h5 X X h8 X X h11 X X X X
h15 -> X X h3 X X h6 X X h9 X X h12 X X X
h16 -> h1 X X h4 X X h7 X X h10 X X h13 X X
*** Results: 70% dropped (70/240 received)
mininet>
```

```
ryu-manager controller.py
pncourse@pncourse-VirtualBox:~/Data-Center-Networking/Project 4 (main)
[17:28:20 2021/05/09] (bash)
$ ryu-manager controller.py
loading app controller.py
loading app ryu.controller.ofp_handler
instantiating app controller.py of SimpleSwitch13
instantiating app ryu.controller.ofp_handler of OFPHandler
```

b. UDP test

B. Difficulties or bottleneck

In this project, I encountered two difficulties which are background traffic and broadcast traffic. After I implemented the first version and started testing, I found that my controller learns unknown MAC addresses, and it cannot pass the UDP test. I classify unknown MAC addresses into the background traffic in the mininet, so I can handle them by using the learning bridge mechanism. UDP test failed because the controller doesn't know how to handle the broadcast traffic for each tenancy. I add new logic to deal with broadcast traffic.

C. Advantage and disadvantage

1. Advantage

Less configuration is needed because the network graph is applied. Only tenancy and leaf-to-host are needed.

2. Disadvantage

Because the controller sends LLDP packets every 5 seconds, it produces extra traffic in the network.