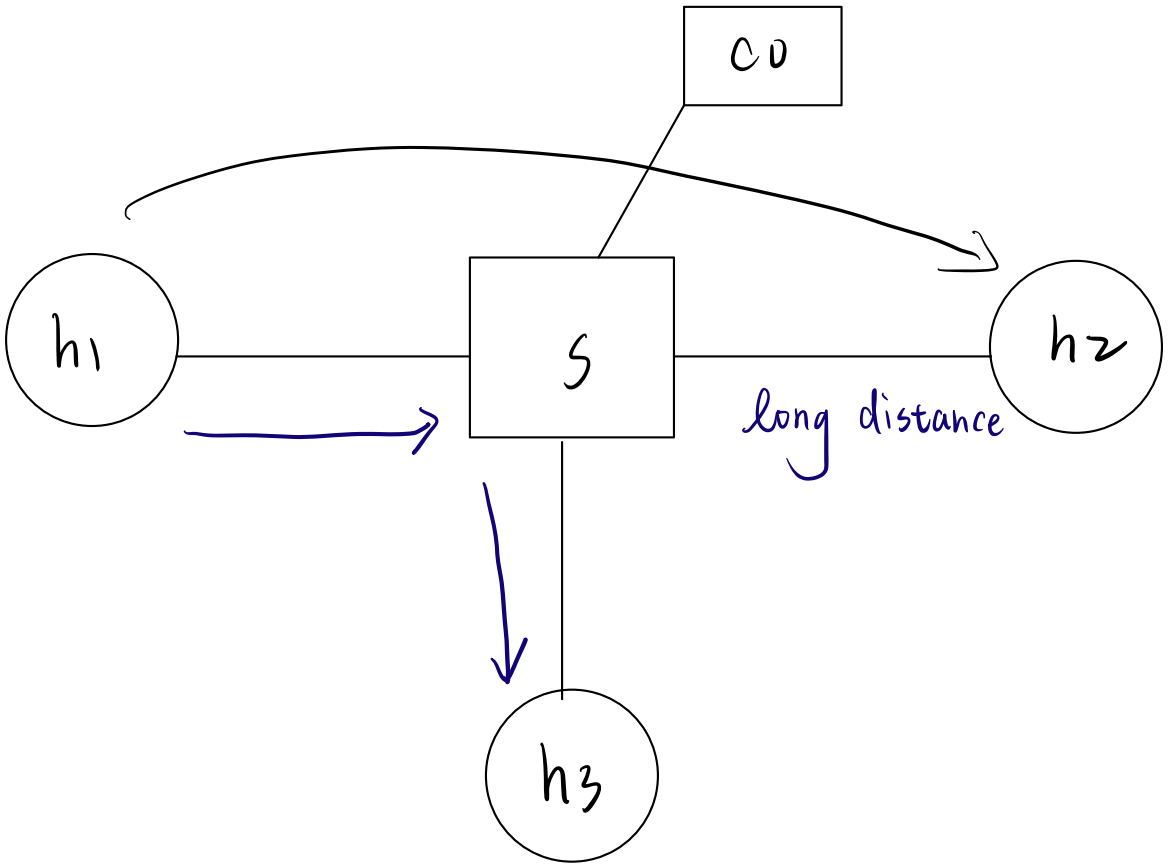
h1 ping h2，但其實封包是送到h3（單向）

運用在DB，索取資料，DNS查詢



/mininet-wifi/example# python3 miniedit.py

設定 C0 Type：Remote Controller

/test # python 4.py

> xterm h1 h2 h3

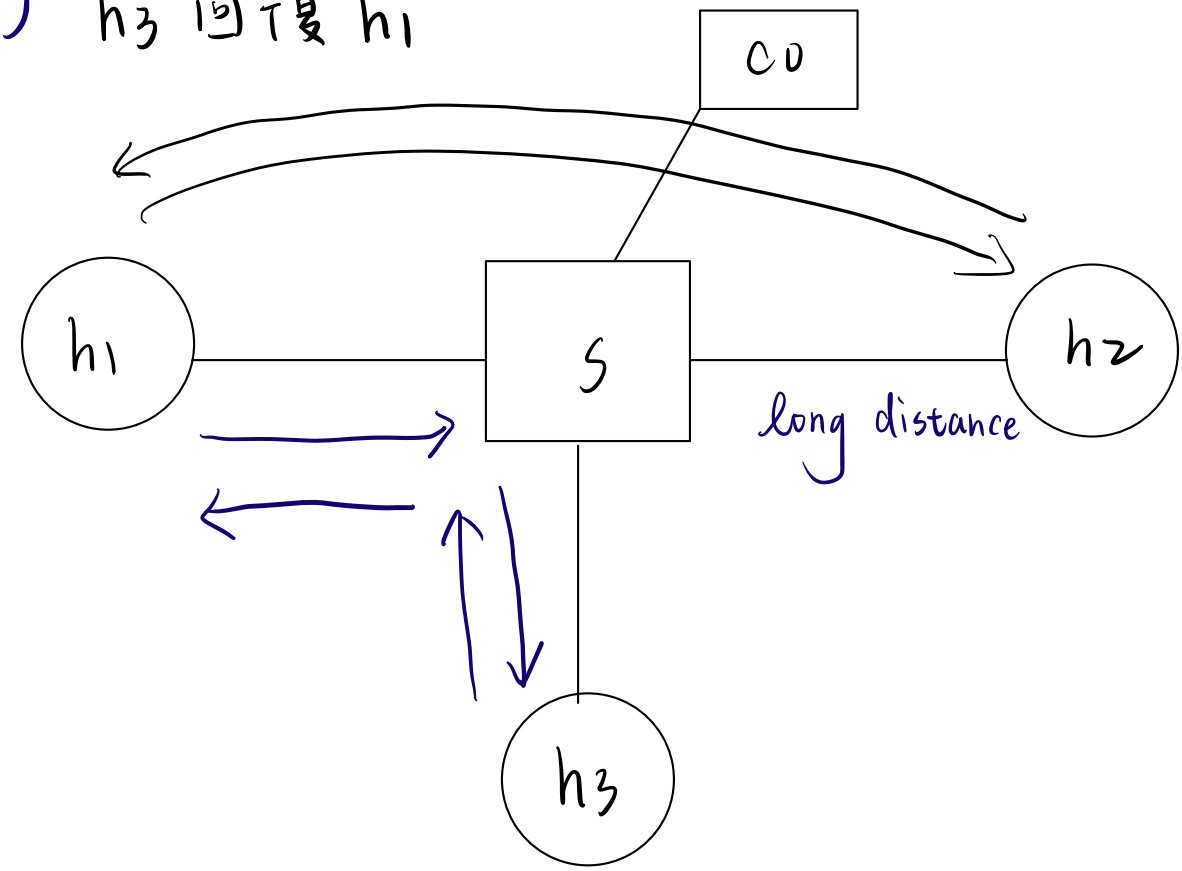
h2
iperf -s -i 1 -u

h3
iperf -s -i 1 -u

h1
iperf -c 10.0.0.2 -u -b 1M -t 100

> sh ovs-ofctl add-flows s1 ip,nw_src=10.0.0.1,nw_dst=10.0.0.2,actions=mod_dl_dst=00:00:00:00:00:03,mod_nw_dst=10.0.0.3,output:3

(双向) h3 回復 h1



> sh ovs-ofctl add-flows s1 ip,nw_src=10.0.0.1,nw_dst=10.0.0.2,actions=mod_dl_dst=00:00:00:00:00:03,mod_nw_dst=10.0.0.3,output:3

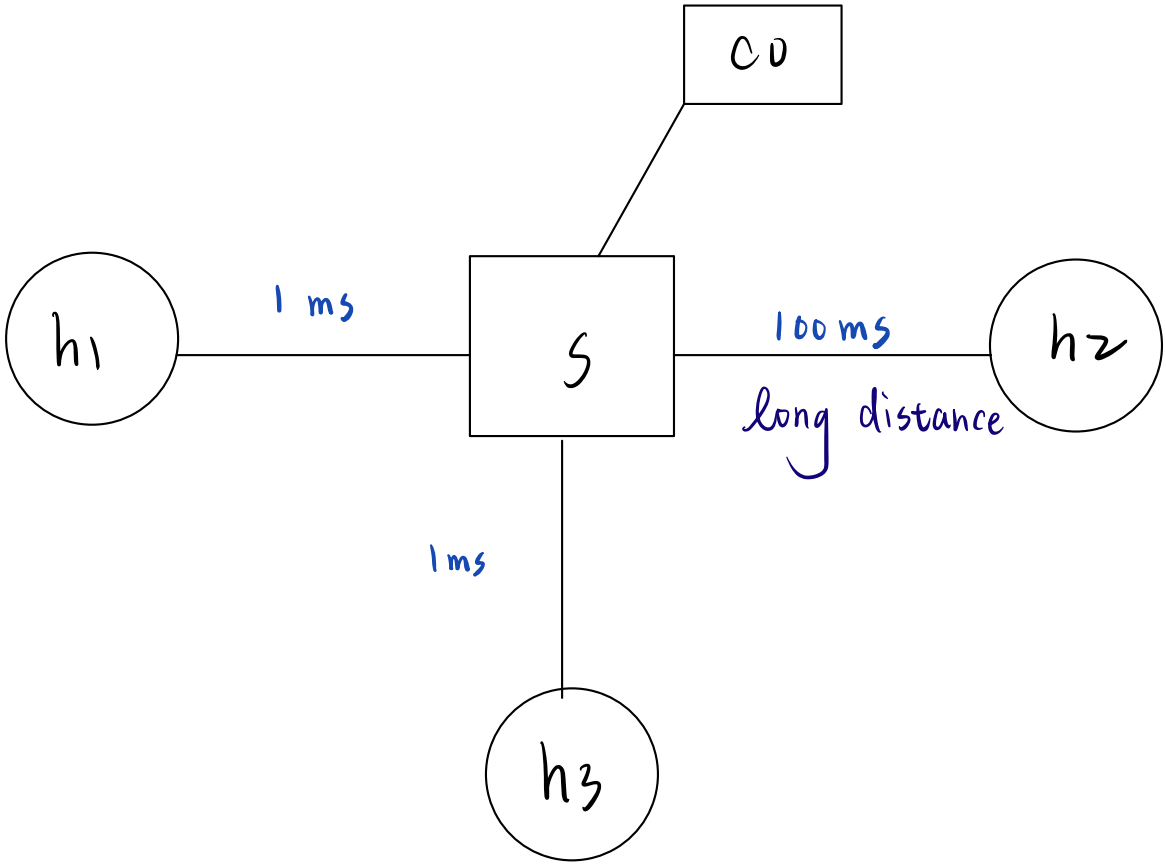
> sh ovs-ofctl add-flows s1 ip,nw_src=10.0.0.3,nw_dst=10.0.0.1,actions=mod_nw_src=10.0.0.2,output:1

> xterm h1 h2 h3

h2
tcpdump -i h2-eth0

h3
tcpdump -i h3-eth0
10.0.0.1 > 10.0.0.3
10.0.0.3 > 10.0.0.1

h1
ping h2



h1 - h2 > 202ms
h1 - h3 > 4 ms

python 4.py

> h1 ping h2 -c 5

206 ms
203
205
:

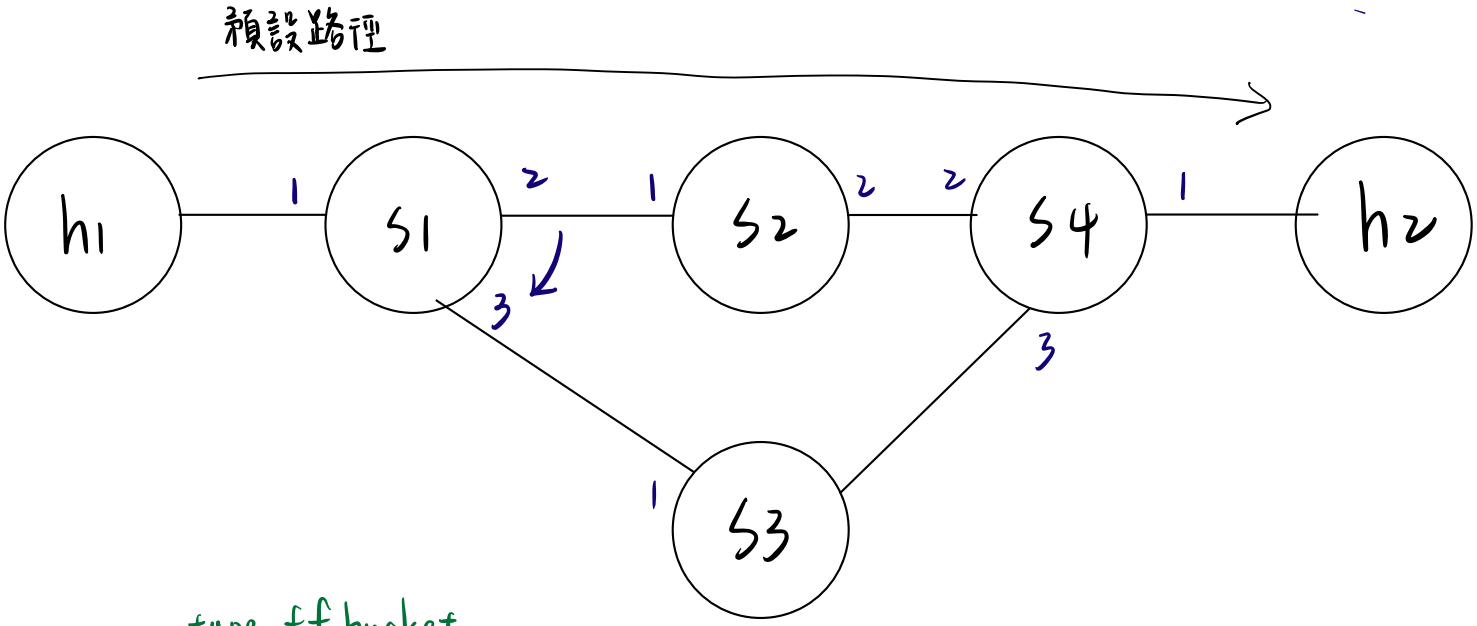
> h1 ping h3 -c 5

6.70 ms
8.11
7.08

link s1 s2 down

openflow 1.3 新功能 1. fastfallover
快速的錯誤切換

當發生錯誤，2 port 切換到 3 port



type=ff bucket
↓
fast-fallover

```
# python test-fast-fallover.py
```

new terminal

```
# chmod +x rule1.sh
# ./rule1.sh
```

```
> sh ovs-ofctl -O OpenFlow13 dump-flows s1
> xterm h1
```

h1

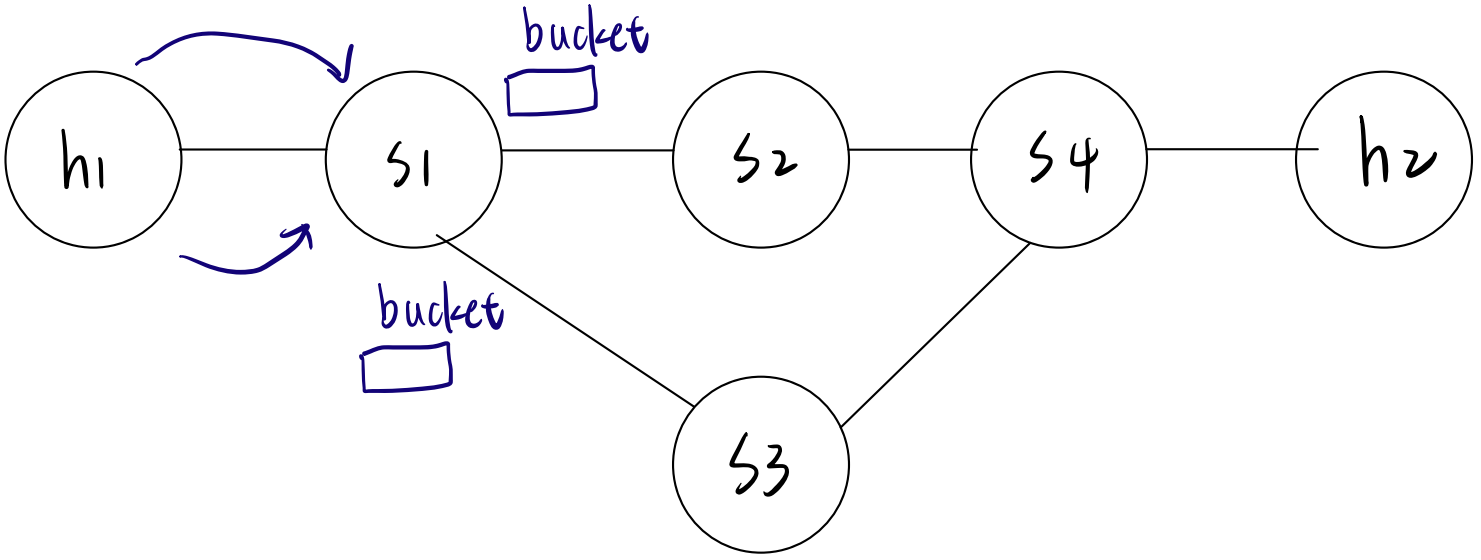
```
# ping 10.0.0.2
```

```
> sh ovs-ofctl dump-port s1
> link s1 s2 down
```

關掉 2 port 路線

```
> sh ovs-ofctl dump-port s1
> link s1 s2 up
```

開啟 2 port 路線

2. group-select : load balancer

隨机挑選 bucket，這樣路徑都会被用到