Lab 7 (Multiple Tables Test)

- In this lab, the Type of Service field of packet will be marked to 0x10 for the first operation in the switch. Then this packet will be sent to the second table for flooding.

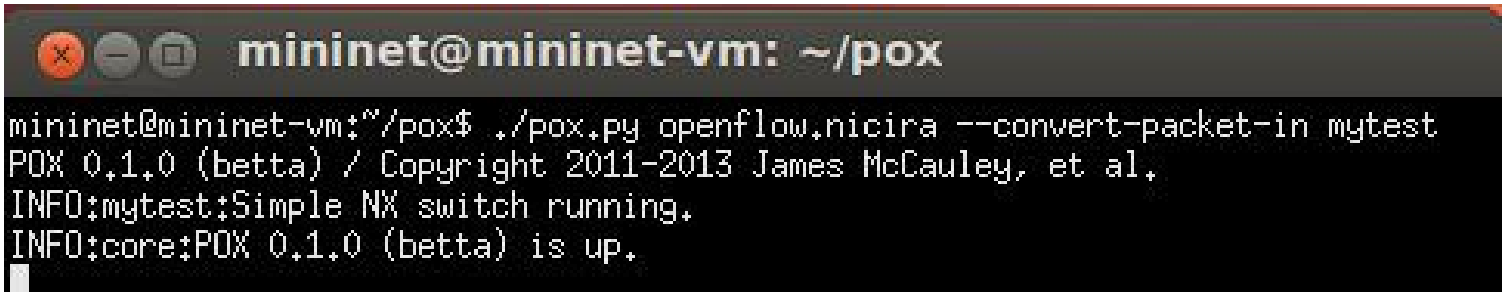
Dr. Chih-Heng Ke
smallko@gmail.com

Put this file, mytest.py, under /pox/ext

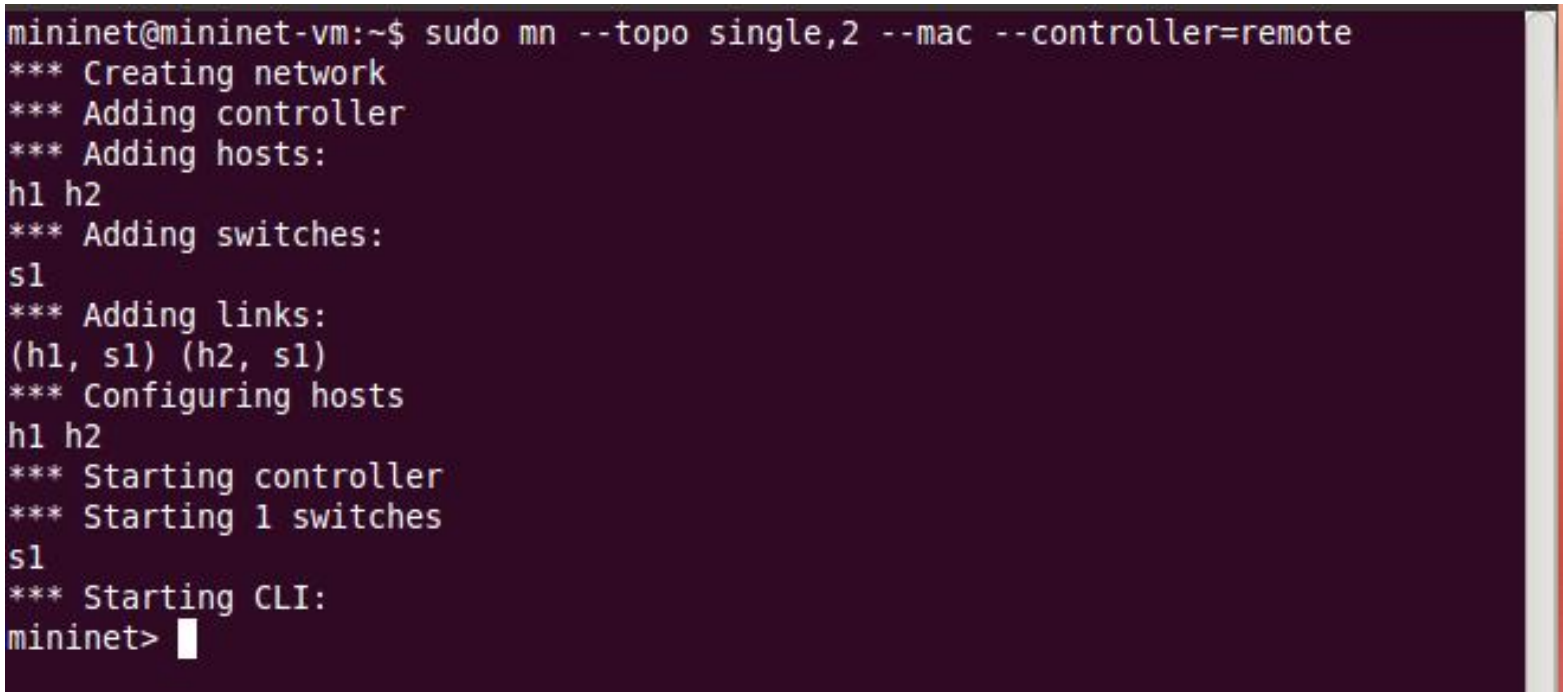
```
from pox.core import core
from pox.lib.addresses import EthAddr
import pox.openflow.libopenflow_01 as of
import pox.openflow.nicira as nx
from pox.lib.revent import EventRemove

# Even a simple usage of the logger is much nicer than print!
log = core.getLogger()

def _handle_ConnectionUp (event):
  print "_handle_ConnectionUP"

  # Turn on ability to specify table in flow_mods
  msg = nx.nx_flow_mod_table_id()
  event.connection.send(msg)

  # Clear second table
  msg = nx.nx_flow_mod(command=of.OFPFC_DELETE, table_id = 1)
  event.connection.send(msg)
```

```
  # Fallthrough rule for table 0: set nw_tos and resubmit to table 1
  msg = nx.nx_flow_mod()
  msg.priority = 1
  msg.actions.append(of.ofp_action_nw_tos(nw_tos=0x10))
  msg.actions.append(nx.nx_action_resubmit.resubmit_table(table = 1))
  event.connection.send(msg)

  # Fallthrough rule for table 1: flood
  msg = nx.nx_flow_mod()
  msg.table_id = 1
  msg.priority = 1
  msg.actions.append(of.ofp_action_output(port = of.OFPP_FLOOD))
  event.connection.send(msg)

def launch ():
  assert core.NX, "Nicira extensions required"
  assert core.NX.convert_packet_in, "PacketIn conversion required"
  core.openflow.addListenerByName("ConnectionUp", _handle_ConnectionUp)

  log.info("Simple NX switch running.")
```

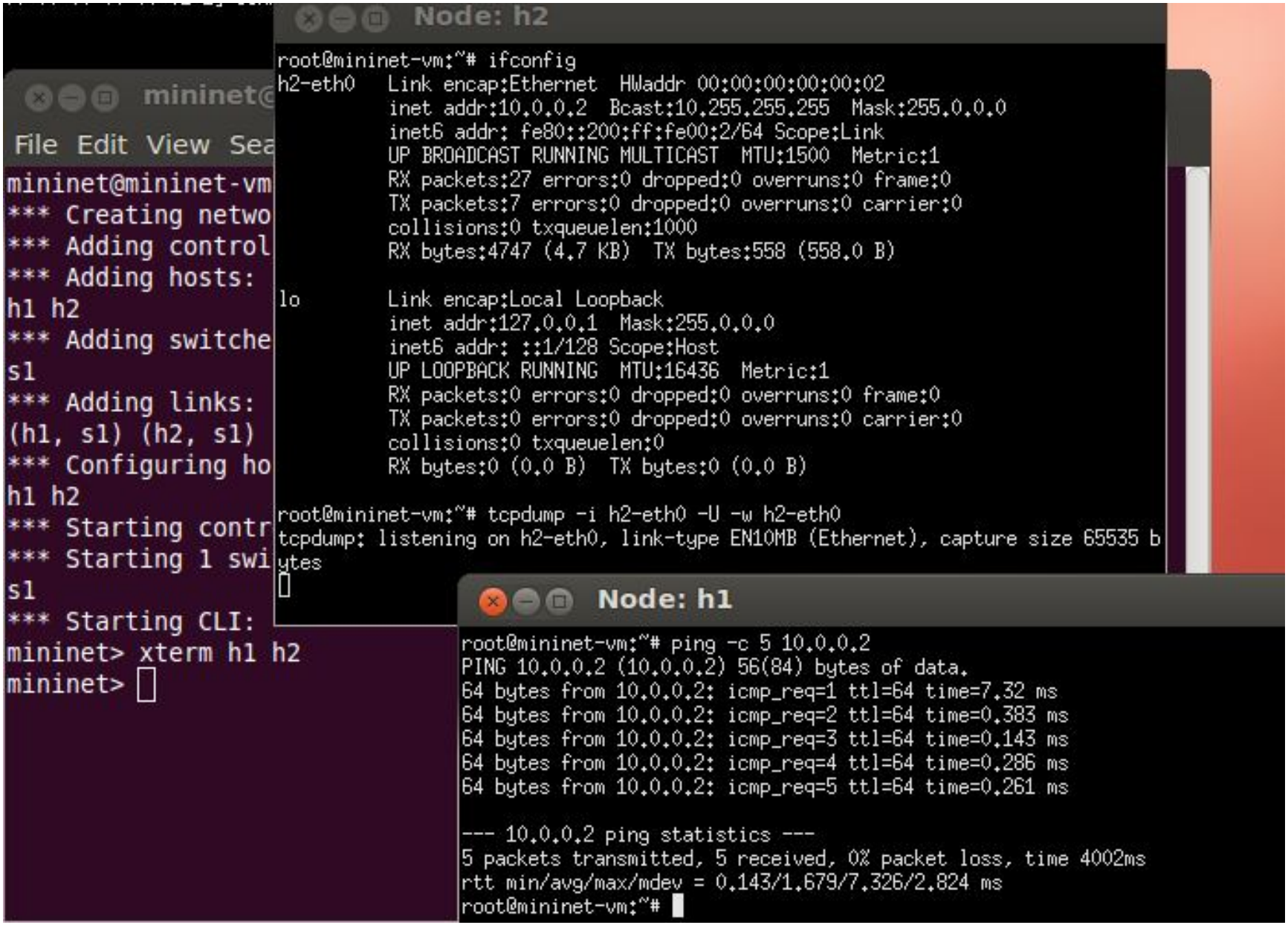
Start the controller

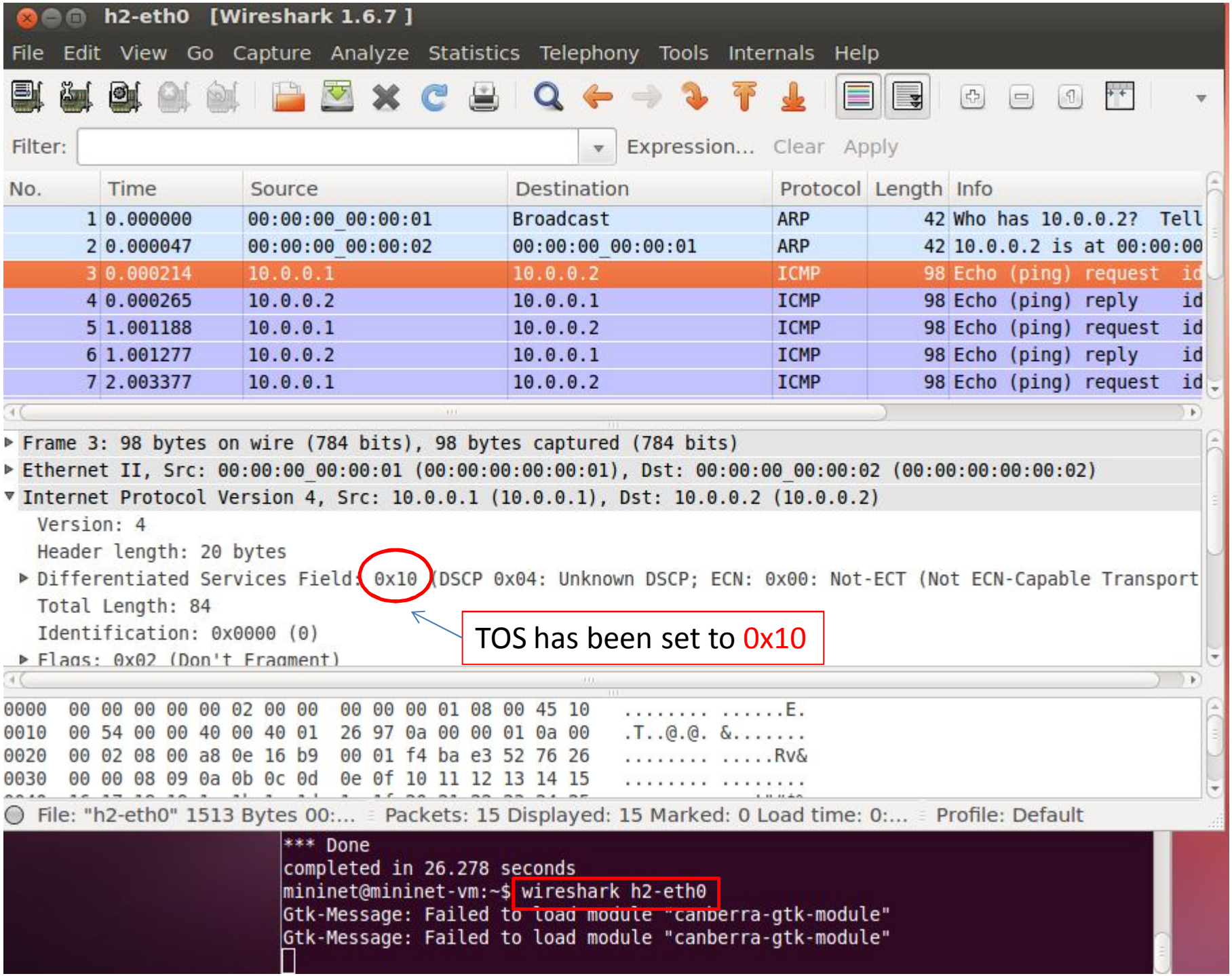


Create a simple topology: one switch and two hosts.



Open xterm for h1 and h2. Use tcpdump in h2 to capture received packets sent by h1




h2-eth0 [Wireshark 1.6.7]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: Expression... Clear Apply
No. Time Source Destination Protocol Length Info
1 0.000000 00:00:00_00:00:01 Broadcast ARP 42 Who has 10.0.0.2? Tell
2 0.000047 00:00:00_00:00:02 00:00:00_00:00:01 ARP 42 10.0.0.2 is at 00:00:00
3 0.000214 10.0.0.1 10.0.0.2 ICMP 98 Echo (ping) request id
4 0.000265 10.0.0.2 10.0.0.1 ICMP 98 Echo (ping) reply id
5 1.001188 10.0.0.1 10.0.0.2 ICMP 98 Echo (ping) request id
6 1.001277 10.0.0.2 10.0.0.1 ICMP 98 Echo (ping) reply id
7 2.003377 10.0.0.1 10.0.0.2 ICMP 98 Echo (ping) request id
Frame 3: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
Ethernet II, Src: 00:00:00_00:00:01 (00:00:00:00:00:01), Dst: 00:00:00_00:00:02 (00:00:00:00:00:02)
Internet Protocol Version 4, Src: 10.0.0.1 (10.0.0.1), Dst: 10.0.0.2 (10.0.0.2)
Version: 4
Header length: 20 bytes
Differentiated Services Field: 0x10 (DSCP 0x04: Unknown DSCP; ECN: 0x00: Not-ECT (Not ECN-Capable Transport
Total Length: 84
Identification: 0x0000 (0)
Flags: 0x02 (Don't Fragment)
TOS has been set to 0x10
0000 00 00 00 00 00 02 00 00 00 00 00 01 08 00 45 10 ....E.
0010 00 54 00 00 40 00 40 01 26 97 0a 00 00 01 0a 00 .T..@.@. &.....
0020 00 02 08 00 a8 0e 16 b9 00 01 f4 ba e3 52 76 26 ....Rv&
0030 00 00 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 ....
File: "h2-eth0" 1513 Bytes 00:... Packets: 15 Displayed: 15 Marked: 0 Load time: 0:... Profile: Default
*** Done
completed in 26.278 seconds
mininet@mininet-vm:~$ wireshark h2-eth0
Gtk-Message: Failed to load module "canberra-gtk-module"
Gtk-Message: Failed to load module "canberra-gtk-module"

References

- POX Wiki, Nicira/Open vSwitch Extensions https://openflow.stanford.edu/display/ONL/POX+Wiki
- The source code under /pox/pox/forwarding.l2_nx.py