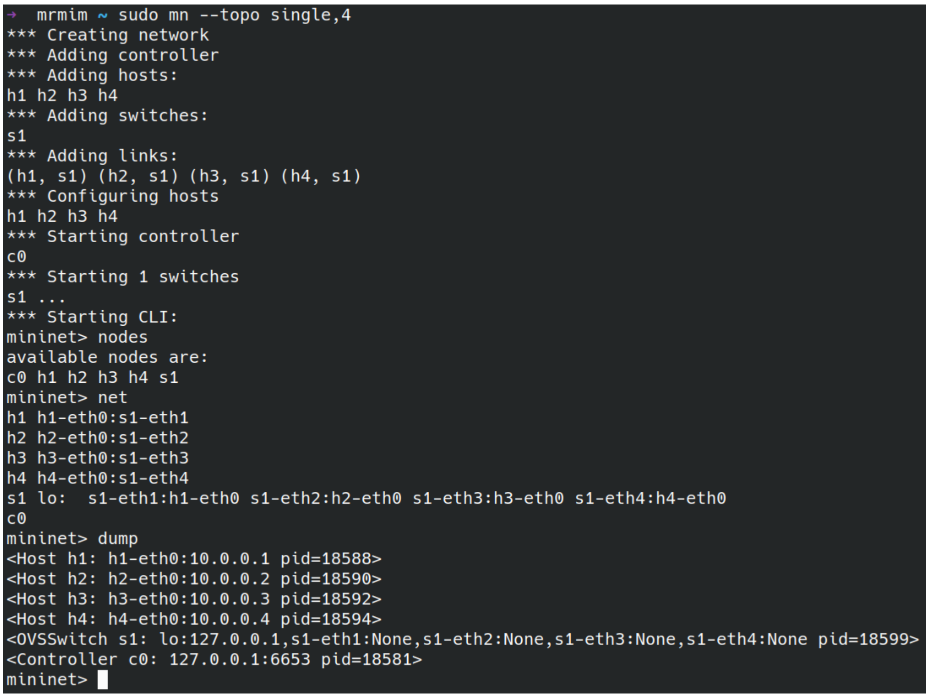
Network 2: Project 1

Dr. Hashemi

December 2023

MohammadMahdi Ahmadi – 9816743

Question 1.1



```
➜  mrmim ~ sudo mn --topo single,4
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2 h3 h4
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1) (h3, s1) (h4, s1)
*** Configuring hosts
h1 h2 h3 h4
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet> nodes
available nodes are:
c0 h1 h2 h3 h4 s1
mininet> net
h1 h1-eth0:s1-eth1
h2 h2-eth0:s1-eth2
h3 h3-eth0:s1-eth3
h4 h4-eth0:s1-eth4
s1 lo:  s1-eth1:h1-eth0 s1-eth2:h2-eth0 s1-eth3:h3-eth0 s1-eth4:h4-eth0
c0
mininet> dump
<Host h1: h1-eth0:10.0.0.1 pid=18588>
<Host h2: h2-eth0:10.0.0.2 pid=18590>
<Host h3: h3-eth0:10.0.0.3 pid=18592>
<Host h4: h4-eth0:10.0.0.4 pid=18594>
<OVSSwitch s1: lo:127.0.0.1,s1-eth1:None,s1-eth2:None,s1-eth3:None,s1-eth4:None pid=18599>
<Controller c0: 127.0.0.1:6653 pid=18581>
mininet> 
```

Question 1.2

`mininet> nodes`: Lists the available nodes in the Mininet network, including the controller (`c0`), hosts (`h1, h2, h3, h4`), and the switch (`s1`).

`mininet> net`: Provides a concise representation of the network topology, showing the connections between hosts and switches. For example, each host (`h1, h2, h3, h4`) is connected to switch (`s1`) through specific interfaces (`h1-eth0:s1-eth1, h2-eth0:s1-eth2, etc.`).

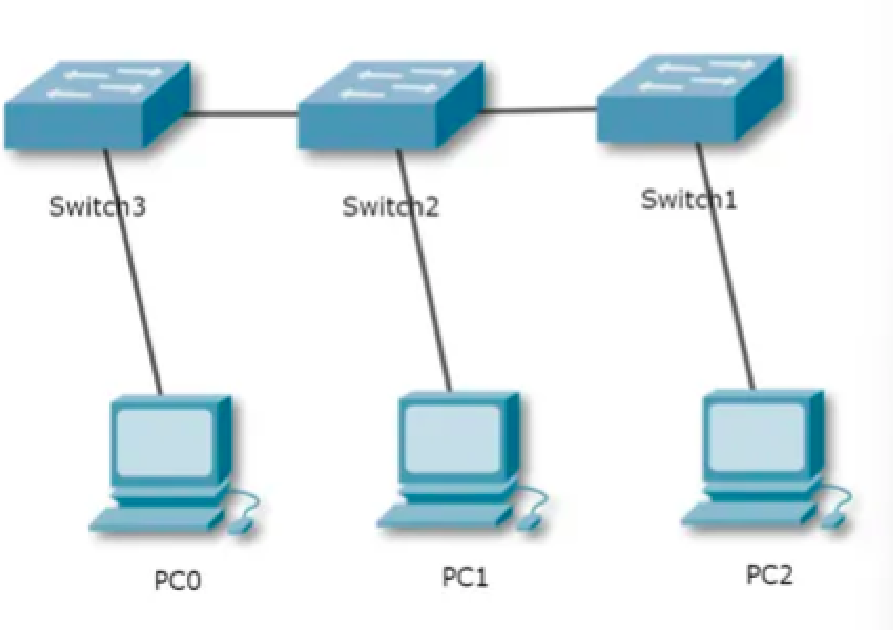
- `h1 h1-eth0:s1-eth1`: Host `h1` is connected to Switch `s1` through interfaces `h1-eth0` and `s1-eth1`.
- `h2 h2-eth0:s1-eth2`: Host `h2` is connected to Switch `s1` through interfaces `h2-eth0` and `s1-eth2`.
- `h3 h3-eth0:s1-eth3`: Host `h3` is connected to Switch `s1` through interfaces `h3-eth0` and `s1-eth3`.
- `h4 h4-eth0:s1-eth4`: Host `h4` is connected to Switch `s1` through interfaces `h4-eth0` and `s1-eth4`.
- `s1 lo:  s1-eth1:h1-eth0 s1-eth2:h2-eth0 s1-eth3:h3-eth0 s1-eth4:h4-eth0`: Describes the loopback interface (`lo`) and the connections between Switch `s1` and hosts (`h1`, `h2`, `h3`, `h4`).
- `c0`: Controller `c0` represents the controller node in the Mininet network.

`mininet> dump`: Provides detailed information about each node in the network, including hosts, switches, and the controller. It includes information such as IP addresses, process IDs (PIDs), and interfaces.

- `Host h1: h1-eth0:10.0.0.1 pid=18588`: Host `h1` has IP address `10.0.0.1`, and its process ID is `18588`.
- `Host h2: h2-eth0:10.0.0.2 pid=18590`: Host `h2` has IP address `10.0.0.2`, and its process ID is `18590`.
- `Host h3: h3-eth0:10.0.0.3 pid=18592`: Host `h3` has IP address `10.0.0.3`, and its process ID is `18592`.
- `Host h4: h4-eth0:10.0.0.4 pid=18594`: Host `h4` has IP address `10.0.0.4`, and its process ID is `18594`.
- `OVSSwitch s1: lo:127.0.0.1, s1-eth1:None, s1-eth2:None, s1-eth3:None, s1-eth4:None pid=18599`: Switch `s1` is an Open vSwitch (OVS) switch with a loopback interface (`lo`) and specific connections.
- `Controller c0: 127.0.0.1:6653 pid=18581`: Controller `c0` has IP address `127.0.0.1`, listens on port `6653`, and its process ID is `18581`.

Question 2.1

Linear Topology:

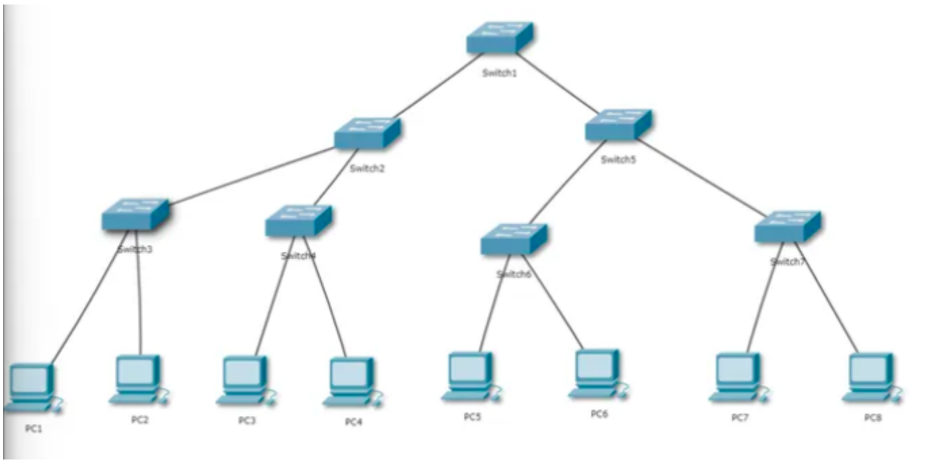


- **Description:** In a linear topology in Mininet, each switch is connected to precisely one host, and all switches are arranged in a straight line. It represents a simple and straightforward network structure resembling a linear chain.
- **Example Command in Mininet:**

  ```
  sudo mn --topo linear,3
  ```

  This command creates a linear topology with three switches, each connected to one host.

Tree Topology:



- **Description:** Tree topology in Mininet is characterized by a hierarchical network structure resembling a tree. The network is organized with a root switch at the top, branching out to child switches at each level. The parameters that define a tree topology include fanout (the number of child switches each switch is connected to) and depth (the depth of the tree).

- **Example Command in Mininet:**

  ```
  sudo mn --topo tree,depth=2,fanout=2
  ```

  This command creates a tree topology with a depth of 2 levels and a fanout of 2 child switches for each parent switch.

Minimal Topology:

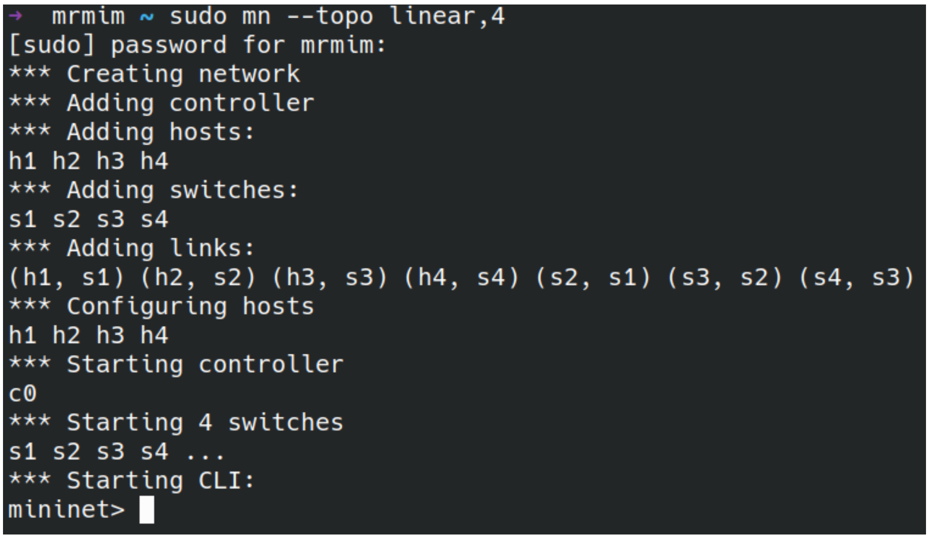
- **Description:** Minimal topology in Mininet is a basic and small-scale network structure that is often used for testing or educational purposes. It typically consists of a minimal set of switches and hosts, providing a simple environment for experimenting with networking concepts.
- **Example Command in Mininet:**

  ```
  sudo mn --topo minimal
  ```

  This command creates a minimal topology with a small number of switches and hosts, suitable for basic testing scenarios.

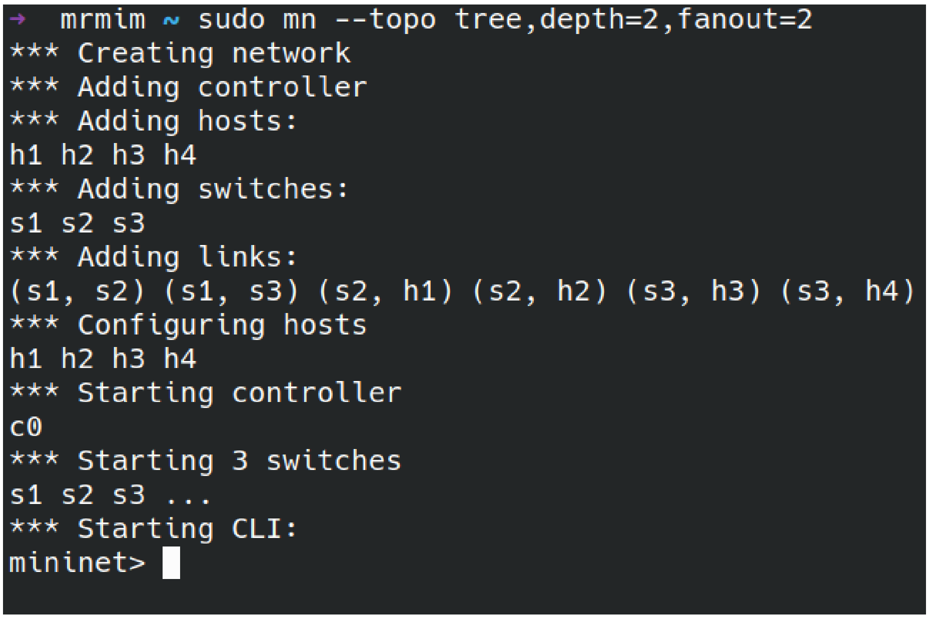
Question 2.2

Linear Topology:



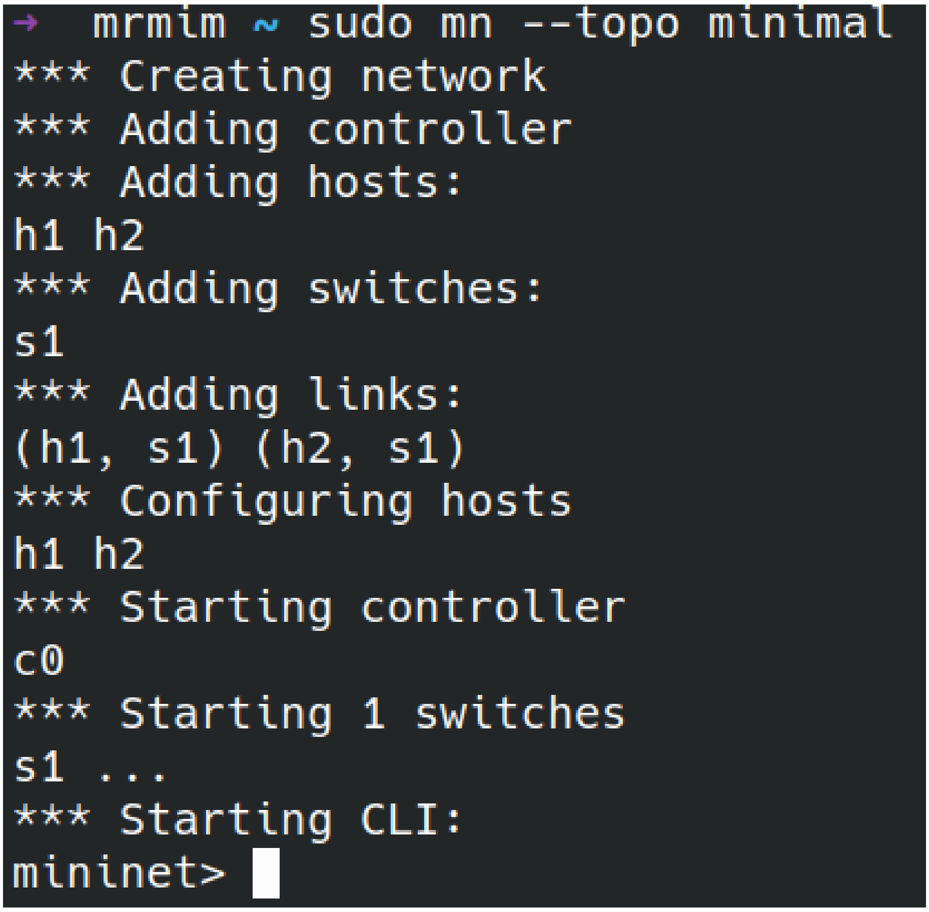
```
➜  mrmim ~ sudo mn --topo linear,4
[sudo] password for mrmim:
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2 h3 h4
*** Adding switches:
s1 s2 s3 s4
*** Adding links:
(h1, s1) (h2, s2) (h3, s3) (h4, s4) (s2, s1) (s3, s2) (s4, s3)
*** Configuring hosts
h1 h2 h3 h4
*** Starting controller
c0
*** Starting 4 switches
s1 s2 s3 s4 ...
*** Starting CLI:
mininet>
```

Tree Topology:



```
➜  mrmim ~ sudo mn --topo tree,depth=2,fanout=2
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2 h3 h4
*** Adding switches:
s1 s2 s3
*** Adding links:
(s1, s2) (s1, s3) (s2, h1) (s2, h2) (s3, h3) (s3, h4)
*** Configuring hosts
h1 h2 h3 h4
*** Starting controller
c0
*** Starting 3 switches
s1 s2 s3 ...
*** Starting CLI:
mininet>
```

Minimal Topology:



```
➜  mrmim ~ sudo mn --topo minimal
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet> 
```

Question 3

`tcpdump` is a powerful packet analyzer that can be used to capture and analyze network traffic. When diagnosing problems in Mininet, `tcpdump` can be a valuable tool to inspect network packets and identify issues. Here are some examples of how you can use `tcpdump` in Mininet:

1. **Start Mininet Topology:** Create a simple Mininet topology, for example:

   ```
   sudo mn −−topo linear ,3
   ```

2. **Open Terminals for Each Host:** Open a terminal for each host in Mininet using the Mininet console.

3. **Run `tcpdump` on a Host:** In the xterm window for a specific host, use the `tcpdump` command to capture and display network traffic. For example, to capture all traffic on host h1:

   ```
   tcpdump −i h1−eth0
   ```

   This command captures traffic on the first interface (`eth0`) of host h1.

4. **Filter Packets:** Use filters with `tcpdump` to capture specific types of packets. For instance, to capture only ICMP (ping) packets on h2:

   ```
   tcpdump −i h2−eth0 icmp
   ```

   This command captures only ICMP packets on the first interface of host h2.

5. **Check for Specific Port Traffic:** You can use `tcpdump` to capture traffic on a specific port. For example, to capture HTTP traffic on h1:

   ```
   tcpdump −i h1−eth0 port 80
   ```

   This command captures packets with destination or source port 80 on the first interface of host h1.