
flokinet-008 - Climbing the Spanning Tree - Lab

Nicholas Morrison nick@nanocat.net



Connecting to the lab server

Open your favourite Terminal Emulator

SSH to the netlab server:

```
$ ssh-keygen -R netlab.nanocat.net <- delete the cached fingerprint
                                (lab server rebuilt frequently)
```

```
$ ssh lab@netlab.nanocat.net
Password: (generated fresh each week)
```

List the running containerlab devices:

```
$ sudo containerlab inspect --all
```

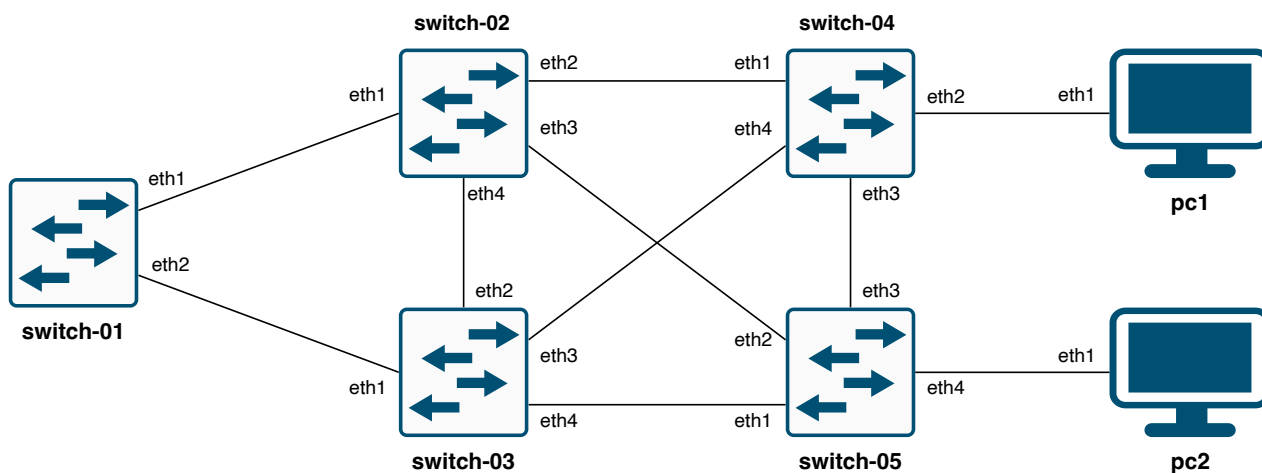
Connect to an **Arista** device:

```
$ sudo docker exec -it clab-device-name Cli
```

.. or connect to a **Linux** device:

```
$ sudo docker exec -it clab-pcXX-name bash
```

Diagram



Goal

- Understand Spanning Tree.

Perform initial configuration

Give all interfaces a description, verified with `show lldp neighbor`.

Do this for all five switches.

```
!
interface ethernet 1
  description --- link to switch-02:eth1 ---
```

```
!  
interface ethernet 2  
    description --- link to switch-03:eth1 ---  
!
```

Record your MAC addresses

Take note of your switches' MAC addresses.

```
switch-01#show spanning-tree
```

Write them down in a text file. You will need them later.

```
switch-01: 01:23:45:67:89:0a  
switch-02: 01:23:45:xx:xx:xx  
switch-03: 01:23:45:xx:xx:xx  
switch-04: 01:23:45:xx:xx:xx  
switch-05: 01:23:45:xx:xx:xx
```

Send a ping

Ping pc2 from pc1.

```
linux pc1#ping 192.168.0.2  
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.  
64 bytes from 192.168.0.2: icmp_seq=1 ttl=64 time=23.9 ms  
64 bytes from 192.168.0.2: icmp_seq=2 ttl=64 time=14.7 ms  
^C  
--- 192.168.0.2 ping statistics ---  
2 packets transmitted, 2 received, 0% packet loss, time 1002ms  
rtt min/avg/max/mdev = 14.680/19.273/23.866/4.593 ms  
linux pc1#
```

Discussion

- Which switch is the **root bridge**?
- Which ports are **designated, root** and **alternate** ports?
- Which layer 2 path did the ping packet follow?
 - Use tcpdump to verify. eg from the CLI: `tcpdump interface eth 3 filter icmp`

Configure the root bridge

```
!  
spanning-tree root primary  
!
```

The default spanning-tree priority is 32768. This command sets the spanning-tree priority to 8192.

Again, check several of your switches by using the `show spanning-tree` command.

- Which switch is the **root bridge**?
- What is the root bridge's priority?
- Which ports are **designated, root** and **alternate** ports?

Control the tree with port costs

Check your current root port on switch-05:

```
switch-05#show span
...
Interface      Role      State      Cost      Prio.Nbr  Type
-----
Et1            alternate discarding 20000     128.1     P2p
Et2            root      forwarding 20000     128.2     P2p
Et3            alternate discarding 20000     128.3     P2p
Et4            designated forwarding 20000     128.4     P2p Edge
```

```
switch-05#
```

Modify the cost of Ethernet 3:

```
!
interface ethernet 3
    spanning-tree cost 10000
!
```

Port costs continued

Check your root port again:

```
switch-05#show span
...
Interface      Role      State      Cost      Prio.Nbr  Type
-----
Et1            alternate discarding 20000     128.1     P2p
Et2            root      forwarding 20000     128.2     P2p
Et3            alternate discarding 10000     128.3     P2p
Et4            designated forwarding 20000     128.4     P2p Edge
```

```
switch-05#
```

Why didn't it change?

How root path cost is calculated

- Root path cost == cost of neighbour's root path, plus interface cost
- show spanning-tree shows the cheapest root cost, and local interface cost

Configure BPDUGuard

BPDUGuard errdisables a port if it receives a BPDU. Enable this on ports connected to devices that should never run spanning tree (eg hosts, routers).

```
! on switch-05
!
interface ethernet 3
    spanning-tree bpduguard enable
!
interface ethernet 4
    spanning-tree bpduguard enable
!
```

Check the results:

```
switch-05#show span
switch-05#show log
switch-05#show interface ethernet 3
switch-05#show interface ethernet 4
switch-05#show interface status errdisabled
```

Use `shutdown + no shutdown` to recover from errdisabled status.

What's portfast??

- spanning-tree portfast edge ports
 - the switch *assumes* that an “edge” port is connected to a single device
 - it transitions immediately to forwarding, and does not check for loops
 - if a BPDU is ever received on an “edge” port, it converts to a “network” port
- spanning-tree portfast network ports
 - the switch *assumes* that a “network” port is connected to another switch
 - it transitions immediately to blocking, and waits for the spanning tree to be built
- spanning-tree portfast normal ports
 - have an unspecified topology
- spanning-tree portfast auto is the default
 - if no BPDU is received within a timeout, it becomes an edge port

Manually configure portfast

```
!
interface ethernet 1
    description --- link to switch-02:eth2 ---
    spanning-tree portfast network
!
```

```
interface ethernet 2
  description --- link to pc1 ---
  spanning-tree portfast edge
!
```