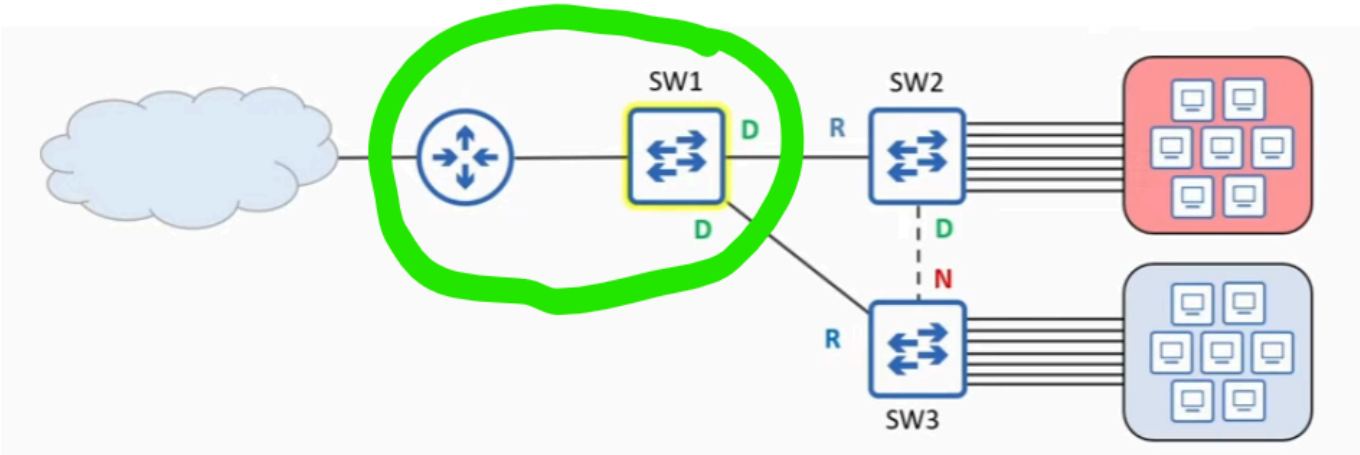
STP Root Guard

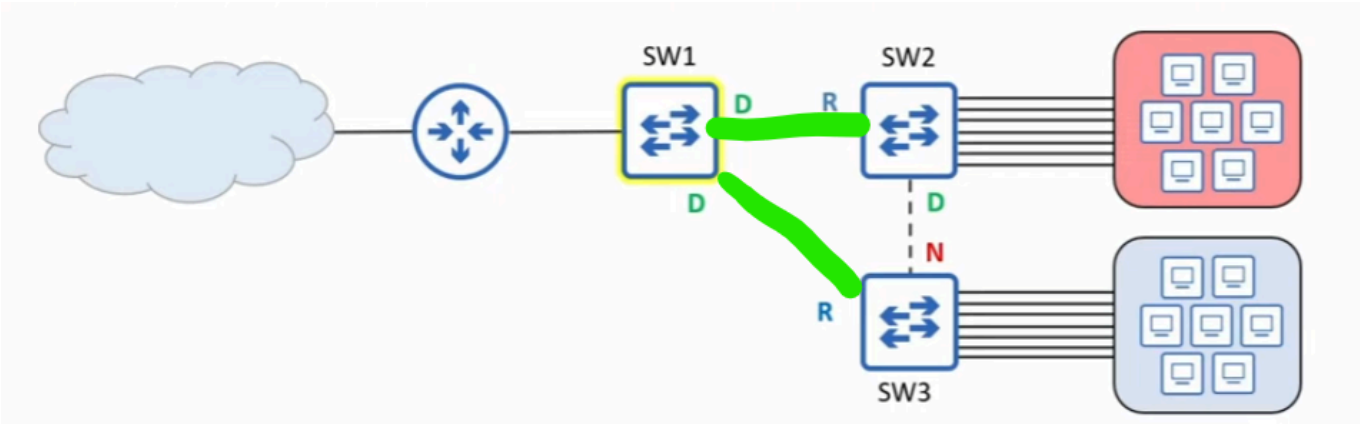
Author - Giovanny Jr Valerio Perdomo

Why Root Bridge Selection Matters

- The **root bridge** is the central reference point for STP. All other switches determine their path to reach it, and redundant links are blocked to prevent loops.
- Selection of the root bridge is not arbitrary; it directly affects network performance.



- **Optimal traffic flow** is a primary consideration. In most LANs, end hosts communicate primarily with servers or the internet, not with each other. Therefore, the root bridge should be placed to provide the most efficient path to the gateway router (e.g., R1).

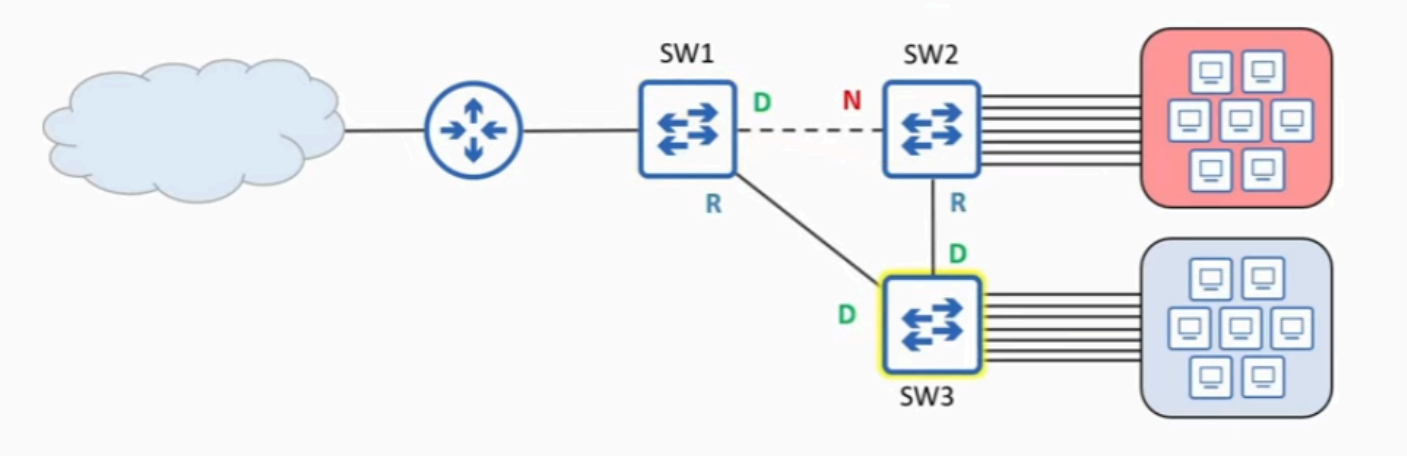


  - Example: In a LAN with SW1, SW2, and SW3, making SW1 the root bridge ensures traffic from SW2 and SW3 takes the direct link to SW1, minimizing **latency** and **congestion**.

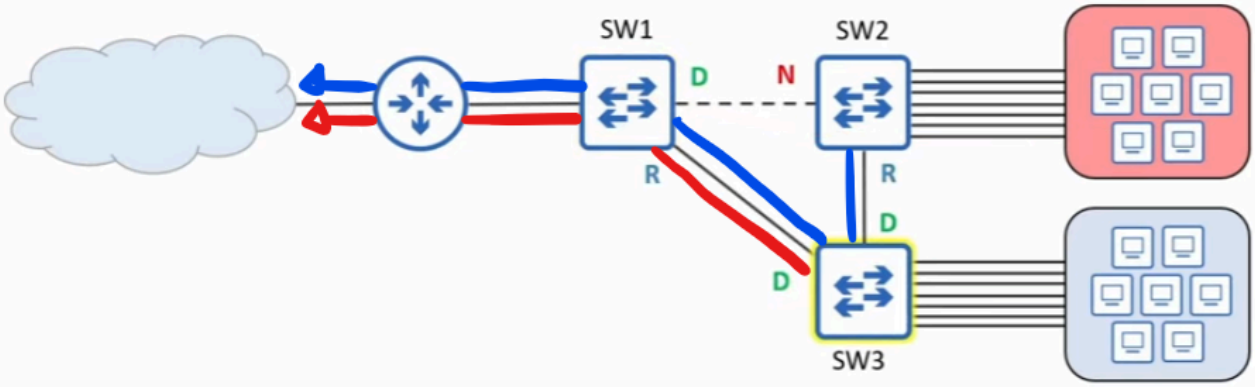
    - **Latency - How much time it takes traffic to travel through the LAN.**
    - **Congestion - How busy the network is; for example, if you try to pump water through a small pipe is not going to be very efficient**.
        - SW2 and SW3 link is disabled to prevent loops.
        - Any of these Switches can be the root bridge, but STP would still fulfill its role in preventing loops.
        - Important: You should not randomly select the root bridge.
        - Good placement of the root bridge is important.
- **Stability and reliability** also matter. The root bridge should be a switch that is robust and unlikely to fail that provides an efficient path for traffic in the LAN, you want that switch to remain up and operational for as long as possible. More advanced or newer switches should be preferred over older, less reliable hardware.

> **Key takeaway:** The root bridge should not be chosen randomly. It should be a stable switch that provides efficient traffic paths.

Consequences of a Poor Root Bridge Selection

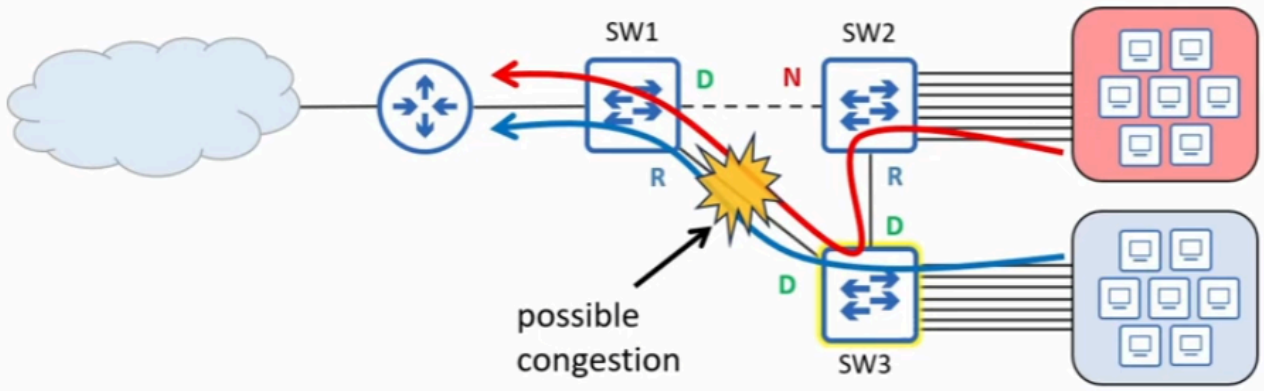


- **Scenario:** SW3 becomes the root bridge instead of SW1.
- **What changes?**



- Traffic from **SW2's hosts** now takes a longer path: **SW2 → SW3 → SW1 → R1**.
- Previously, SW2's traffic went directly to SW1.

- **Resulting issues:**

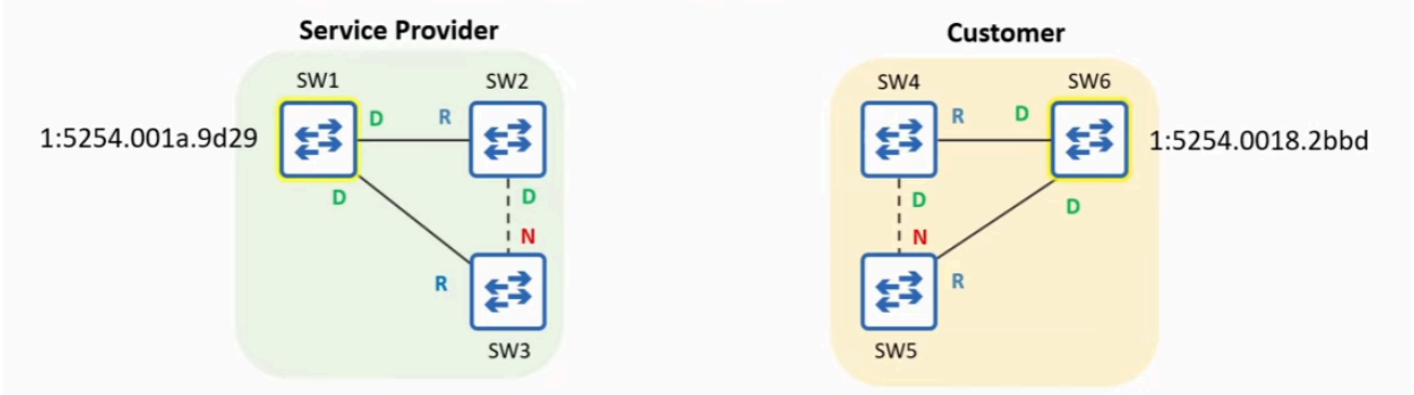


- **Latency** increases slightly (usually less than 1 ms), but this is often negligible.
- **Congestion** becomes the real problem — the **SW3–SW1 link** may become overloaded.
  - When congested, frames must wait to be transmitted.
  - In worst cases, **frames may be dropped**.
  - This degrades network performance and user experience significantly.

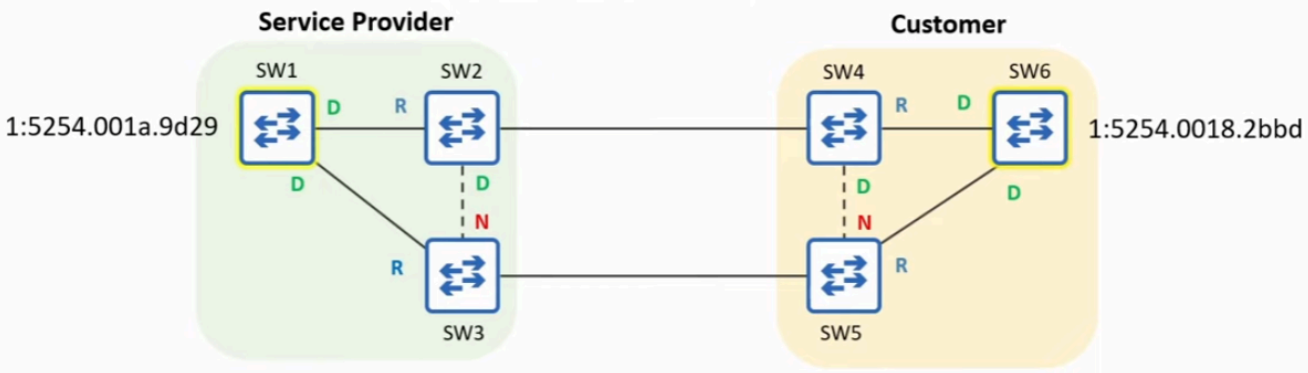
> ***Key takeaway: The root bridge must be carefully chosen. After selection, it should remain stable to avoid these problems.***

The Problem: External Switches Can Disrupt the Root Bridge

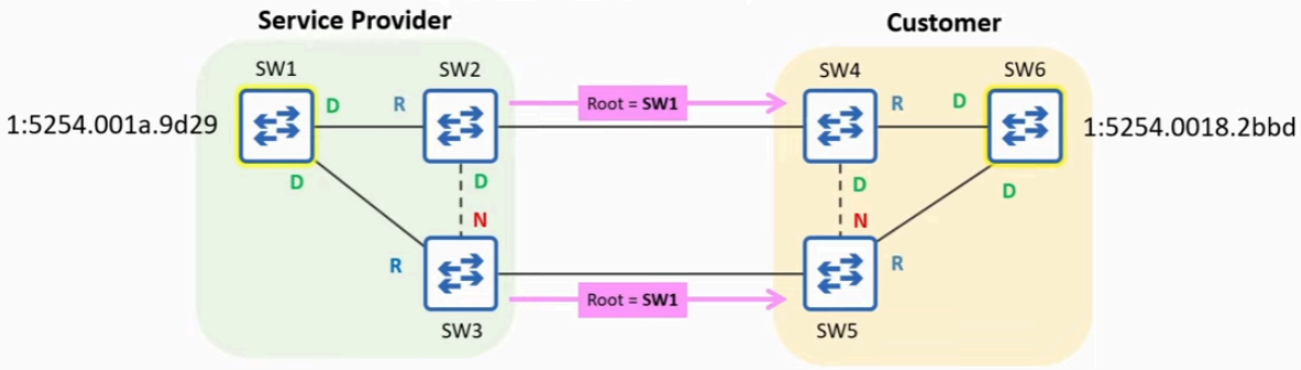
- **Controlling the root bridge within your LAN:** You can easily enforce a specific root bridge by setting its priority to 0 using the command `spanning-tree vlan 1 priority 0`. This ensures the lowest bridge ID (priority + VLAN ID + MAC address) among your switches.
    - Cisco switches run **PVST+** (Per-VLAN Spanning Tree Plus), so the VLAN ID is always added to the priority value.



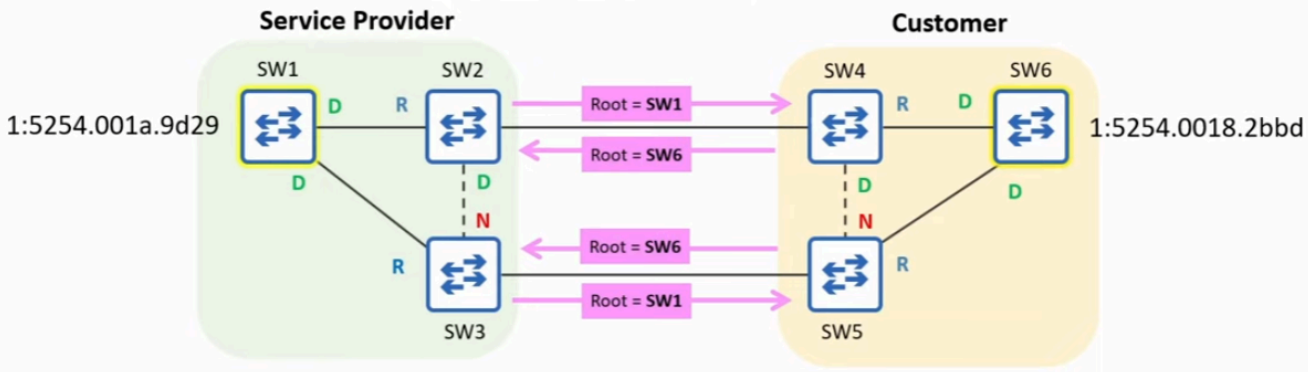
- **Risk with multi-organizational connections:** When connecting your LAN to another organization's network (e.g., a service provider connecting to a customer's network), you lose control over the root bridge election.
    - A service provider offering Metro Ethernet Service to customers
        - Often used to connect sites within a MAN.
- **Example Scenario:**



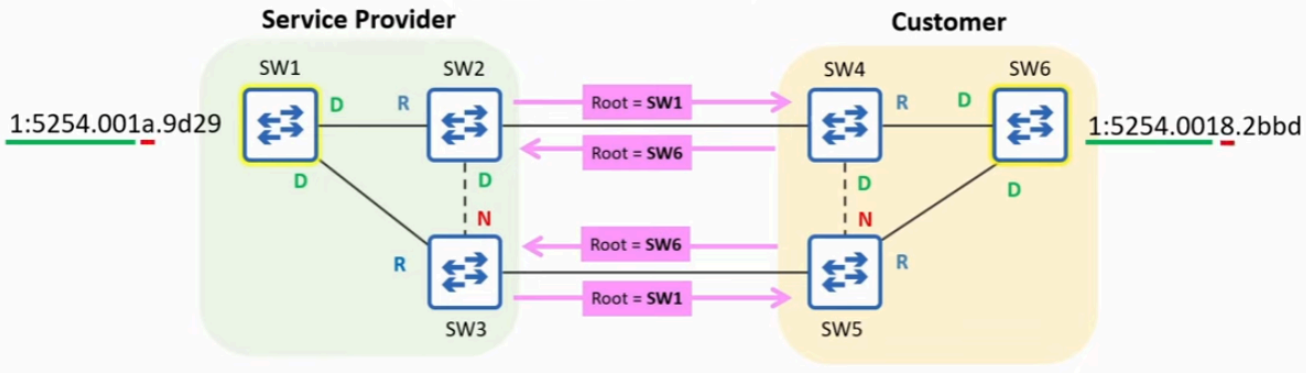
- **Service Provider LAN (left):** SW1 has priority 1 (0 + VLAN 1), MAC address ending in "...a".
- **Customer LAN (right):** SW6 also has priority 1 (0 + VLAN 1), but with a MAC address ending in "...8".



- SW2 and SW3 will send BPDUs saying that SW1 is the root bridge.

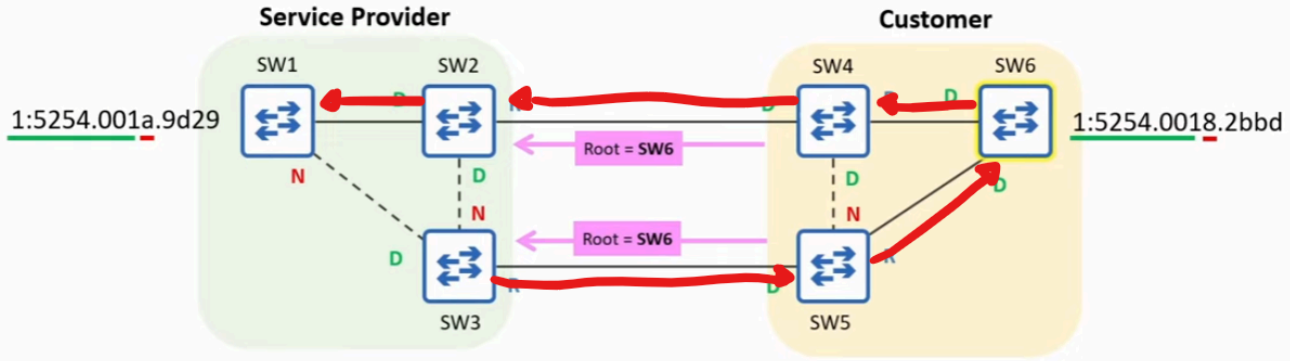


- But SW4 and SW5 will also send BPDU's saying that SW6 is the root bridge.
- Which Switch becomes the root bridge?



- **Outcome:** SW6 has a lower MAC address → lower bridge ID → becomes the root bridge for the entire combined network.

- **Impact of root bridge takeover:**



- SW1, SW2, and SW3 accept SW6 as the root bridge.
- The SW1–SW3 and SW2–SW3 links are disabled.
- Traffic from SW3 to SW1 must detour through the customer's LAN, causing **inefficient paths** and potential congestion.

> ***Key takeaway: Even with priority set to 0, a switch with a lower MAC address can usurp the root bridge role. This can severely disrupt your STP topology when connecting to external networks.***

Root Guard Solution

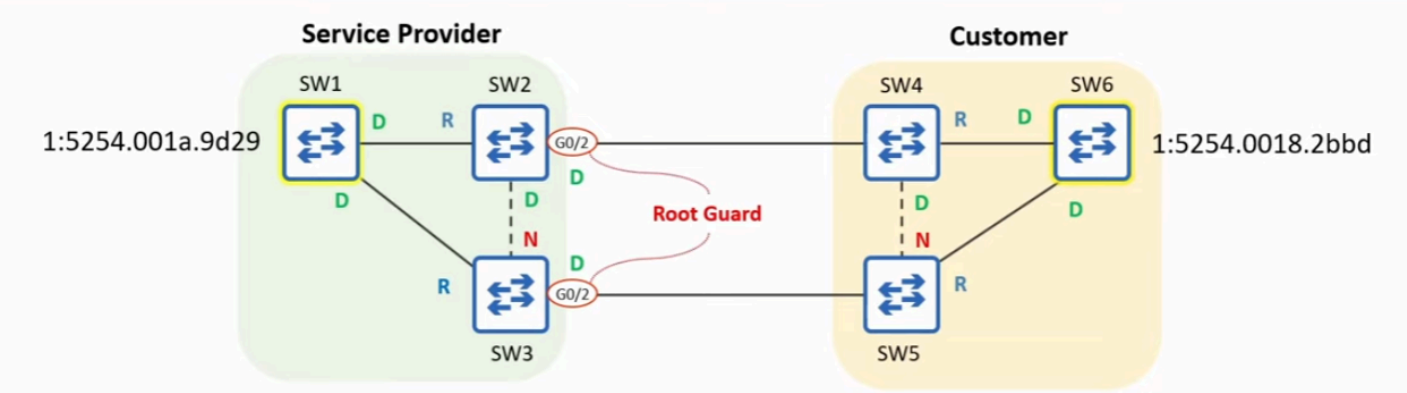
- **Root Guard** is a Cisco STP feature that protects your STP topology by preventing switches from accepting **superior BPDUs** from switches outside your control.
  - A **superior BPDU** is one with better STP parameters (e.g., a lower root bridge ID).

- **Where to apply:** Configure Root Guard on ports that connect to switches outside your administrative control (e.g., service provider ports facing the customer).
- **How it works:**
  - If a Root Guard-enabled port receives a superior BPDU (claiming a better root bridge), the port is **disabled** (enters a "broken" state).
  - This prevents the external switch from becoming the root bridge, **enforcing your intended root bridge**.

> ***Key definition: Root Guard disables a port that receives a superior BPDU, preventing that port from becoming a root port and ensuring the current root bridge remains stable.***

Configuring Root Guard

To ensure the root bridge remains within your LAN, configure Root Guard on ports connected to switches outside your control.

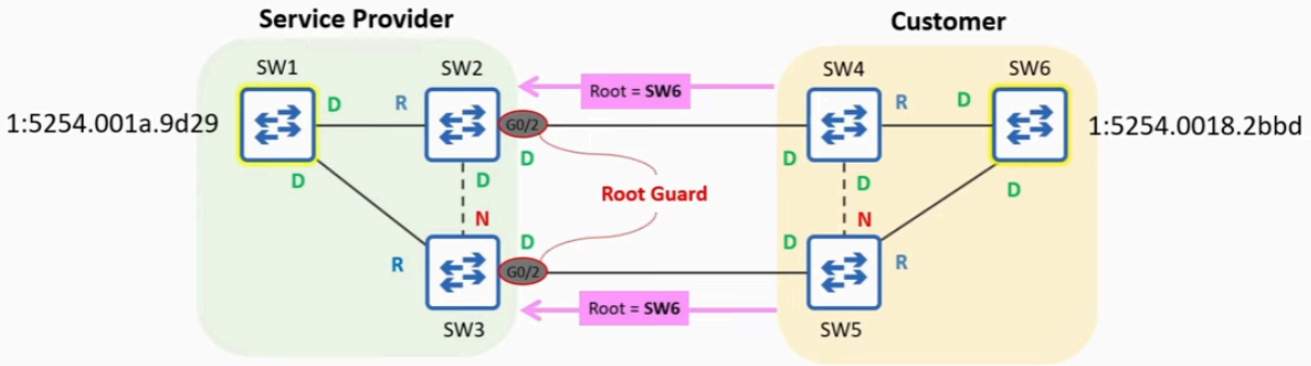


- **Example – Service Provider Setup:**
  - Apply Root Guard on **SW2 G0/2** and **SW3 G0/2**, which connect to the customer's switches.
- **Command:**
  - `spanning-tree guard root` — configured in **interface configuration mode**.
  - Unlike PortFast, BPDU Guard, or BPDU Filter, there is **no global default command** — configuration is **per interface only**.

Root Guard Behavior Upon Receiving Superior BPDUs

When a Root Guard-enabled port receives a BPDU claiming a better root bridge:

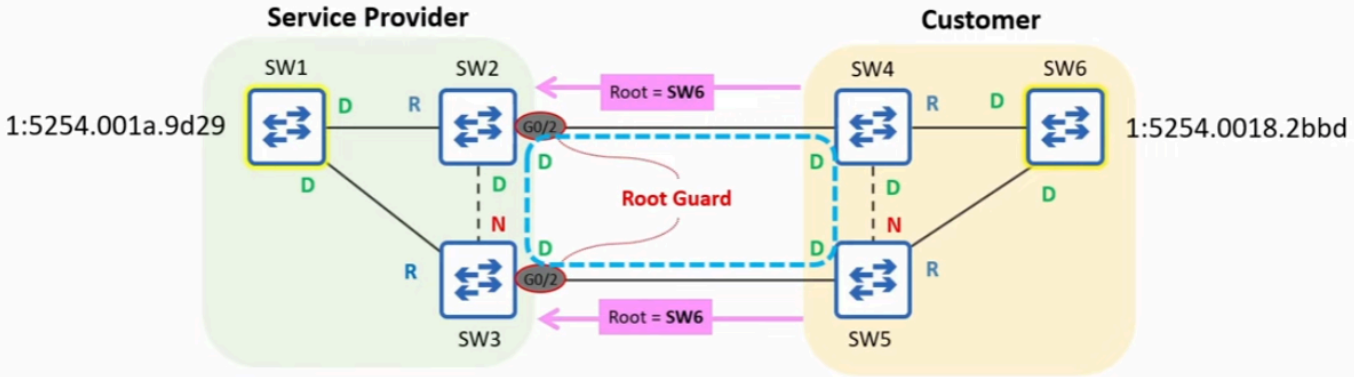
- The port enters a **"broken" (BKN)** or **"root inconsistent"** state.
- The port is **effectively disabled**:
  - Cannot forward data frames.
  - Discards all received frames.
  - All traffic over that link is cut off.
-



Result for the Service Provider:

- SW1, SW2, and SW3 **reject** SW6 as the root bridge.
- The provider's STP topology is protected.
- Despite the physical connection, the provider and customer LANs **cannot communicate** over that link.

Important Note on Port Roles



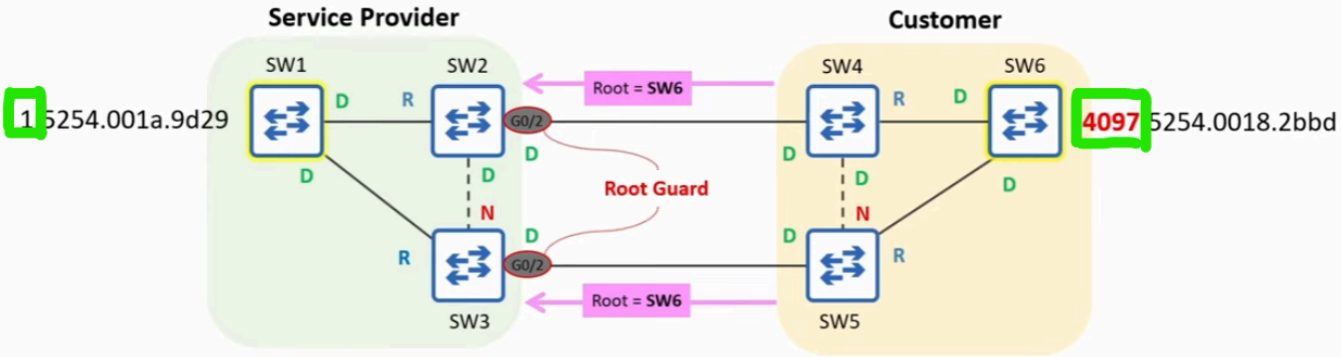
- In this scenario, **SW2 G0/2**, **SW3 G0/2**, and the corresponding ports on SW4 and SW5 are all **designated ports**.
- **Normal STP rule:** Only one designated port per link.
- **Exception here:** The switches disagree on the root bridge, so Root Guard blocks the link — resulting in two designated ports (one on each side) on a non-functional link.

> ***Key takeaway: Root Guard is configured per interface with*** `spanning-tree guard root`***. When a superior BPDU is received, the port is disabled (broken/root inconsistent), cutting all traffic and preventing the external switch from becoming the root bridge.***

Recovery from Root Guard Blocking

Root Guard prevented the customer from influencing the service provider's STP topology — but this also means the customer **cannot communicate** over the service provider's network. How do we fix this?

- **To re-enable a port disabled by Root Guard:** You must solve the issue that disabled the port.
  - In other words, the disabled port needs to **stop receiving superior BPDUs**.
- **Solution (from the provider's perspective):**



  - The service provider tells the customer to **increase the priority of their switch** (SW6), making its bridge ID **inferior** to SW1's.
- **Example fix on the customer's switch (SW6):**

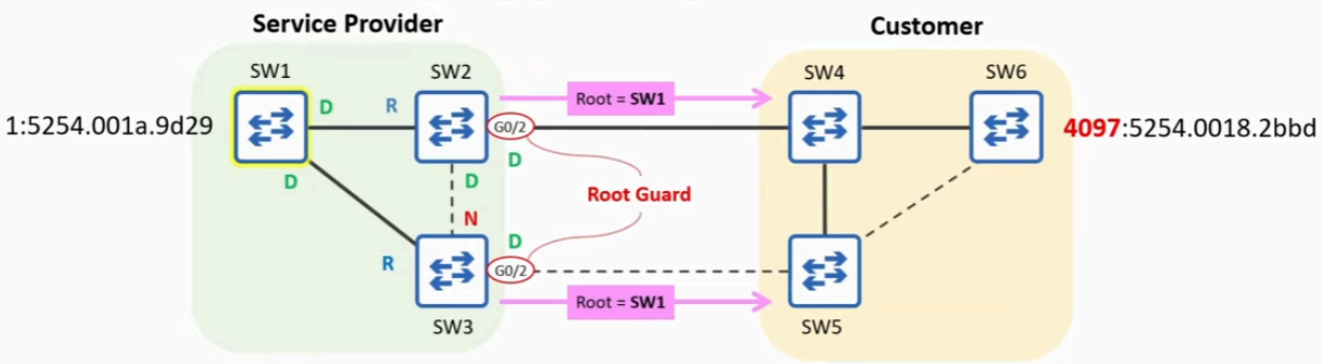
```
SW6(config)# spanning-tree vlan 1 priority 4096
```

- This sets SW6's priority to 4096, which becomes **4097** when the VLAN ID (1) is added.
- SW1's priority of (0 + 1 = 1) is now **superior**, so SW1 becomes the root bridge.

Automatic Recovery Process

Once the superior BPDUs received by SW2 G0/2 and SW3 G0/2 age out, the ports will automatically be re-enabled:

- The ports on SW2 and SW3 (SW2 G0/2 and SW3 G0/2) **automatically recover**.
  - **No manual intervention** is needed on SW2 or SW3 (the switches using Root Guard).
  - No `errdisable recovery` commands are required (unlike BPDU Guard).
- **Timing:**
  - Recovery takes approximately **20 seconds** (the default **Max Age** timer for BPDUs).
  - Once the superior BPDUs age out, the ports return to normal STP operation.
- **Resulting topology:**



  - SW4, SW5, and SW6 accept SW1 as the root bridge.
  - SW1 is the root bridge for the entire combined network.
  - All other switches have one active path to reach SW1.
  - The remaining redundant links are blocked.

> ***Key takeaway: To re-enable a port disabled by Root Guard, you do not need to modify the switch running Root Guard. You simply stop the superior BPDUs from arriving (e.g., by increasing the external switch's priority), and the disabled ports recover automatically after about 20 seconds.***

CLI Demonstration of Root Guard Configuration and Behavior

Since Root Guard only requires a single command to configure, the CLI demonstration is straightforward.

- So once again, SW6 STP priority is 1 making its bridge ID superior to SW1's.

Configuring Root Guard on SW2 (Service Provider)

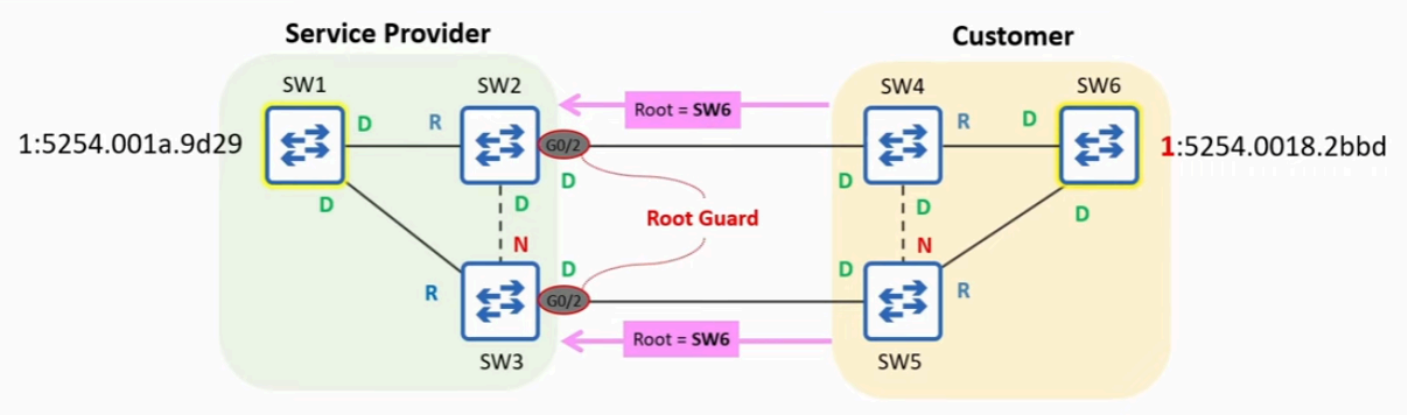
```
SW2(config)# interface g0/2
SW2(config-if)# spanning-tree guard root
```

```
SW2(config)# interface g0/2
SW2(config-if)# spanning-tree guard root
*Sep 21 08:38:56.314: %SPANTREE-2-ROOTGUARD_CONFIG_CHANGE: Root guard enabled on port GigabitEthernet0/2.
```

- A log message confirms: **Root Guard has been enabled on the port.**
- The same configuration is applied to **SW3 G0/2**.

Viewing the Root Guard Blocking State

- So what happens when SW2 and SW3 receive BPDUs from SW5 and SW6?



When SW2 and SW3 receive BPDUs from SW4 and SW5 (claiming SW6 as root with superior bridge ID):

```
SW2(config)# interface g0/2
SW2(config-if)# spanning-tree guard root
*Sep 21 08:38:56.314: %SPANTREE-2-ROOTGUARD_CONFIG_CHANGE: Root guard enabled on port GigabitEthernet0/2.
*Sep 21 08:38:56.321: %SPANTREE-2-ROOTGUARD_BLOCK: Root guard blocking port GigabitEthernet0/2 on VLAN0001.
```

- A log message appears: **Root Guard blocks the ports.**

Use `show spanning-tree` to verify the port status:

```
SW2(config-if)# do show spanning-tree
!output omitted
Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/0               Root FWD 4         128.1    P2p
Gi0/1               Desg FWD 4         128.2    P2p
Gi0/2               Desg BKN*4         128.3    P2p *ROOT_Inc
```

```
SW2# show spanning-tree
```

- **G0/2's status column:** `BKN`
- **Right column:** `ROOT_Inc`

Term	Meaning
`BKN`	**Broken** — the port is disabled; cannot forward or receive data frames
`ROOT_Inc`	**Root Inconsistent** — indicates the port was disabled by **Root Guard**

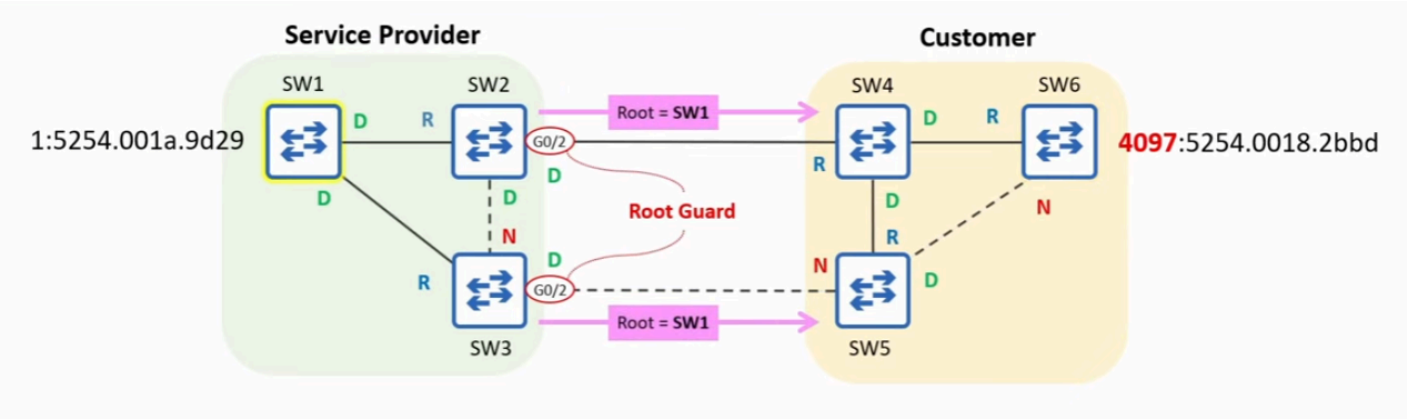
Recovery After Fixing the Issue

After the customer increases **SW6's priority to 4097** (making SW1 the superior root bridge):

- After approximately **20 seconds**, SW2 shows a log message: **G0/2 is now unblocked.**
- Running `show spanning-tree` again confirms:

```
*Sep 21 08:54:26.955: %SPANTREE-2-ROOTGUARD_UNBLOCK: Root guard unblocking port GigabitEthernet0/2 on VLAN0001.
SW2(config-if)# do show spanning-tree
!output omitted
Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi0/0               Root FWD 4         128.1    P2p
Gi0/1               Desg FWD 4         128.2    P2p
Gi0/2               Desg FWD 4         128.3    P2p
```

  - Status is **no longer broken**.
  - The **Root Inconsistent** message is **gone**.



- The network converges once again with **SW1 as the root bridge**.

Where to Configure Root Guard — Important Clarification

- **Correct placement:** Root Guard should be configured on ports **connecting to external switches** that you don't control.
  - Example: Service provider configures Root Guard on **SW2 G0/2** and **SW3 G0/2** (facing the customer's network).
- **What NOT to do:**
  - The **customer** should **not** configure Root Guard on their ports connecting to the provider.
  - If the customer enables Root Guard on SW4 and SW5's ports facing the provider, those ports will be **disabled** upon receiving superior BPDUs from the provider's network.
  - This would block all communication with the service provider — defeating the purpose of connecting to their network.

> ***Key takeaway: Root Guard is configured on the ports facing external switches by the organization that wants to protect its own root bridge. The external organization should not use Root Guard on the connecting ports, or it will block all communication.***

Summary and Comparison with Other STP Toolkit Features

Feature	Global Default?	Behavior
PortFast	Yes (`spanning-tree portfast default`)	Immediately transitions access ports to forwarding; skips listening/learning.

BPDU Guard	Yes (on PortFast-enabled ports via `spanning-tree portfast bpduguard default`)	Disables a PortFast port that receives any BPDU (errdisable).
BPDU Filter	Yes (on PortFast-enabled ports via `spanning-tree portfast bpdufilter default`)	Suppresses BPDU transmission/reception on PortFast ports.
Root Guard	**No** — only per interface (`spanning-tree guard root`)	Disables a port if a superior BPDU is received; automatically recovers when BPDUs stop.

- Root Guard is intentionally not available as a global default because it should only be applied selectively on specific inter-network links, unlike PortFast, which is safe on most access ports.

Root Guard is enforced at the port level and recovers automatically, ensuring the root bridge remains stable without manual re-enabling.