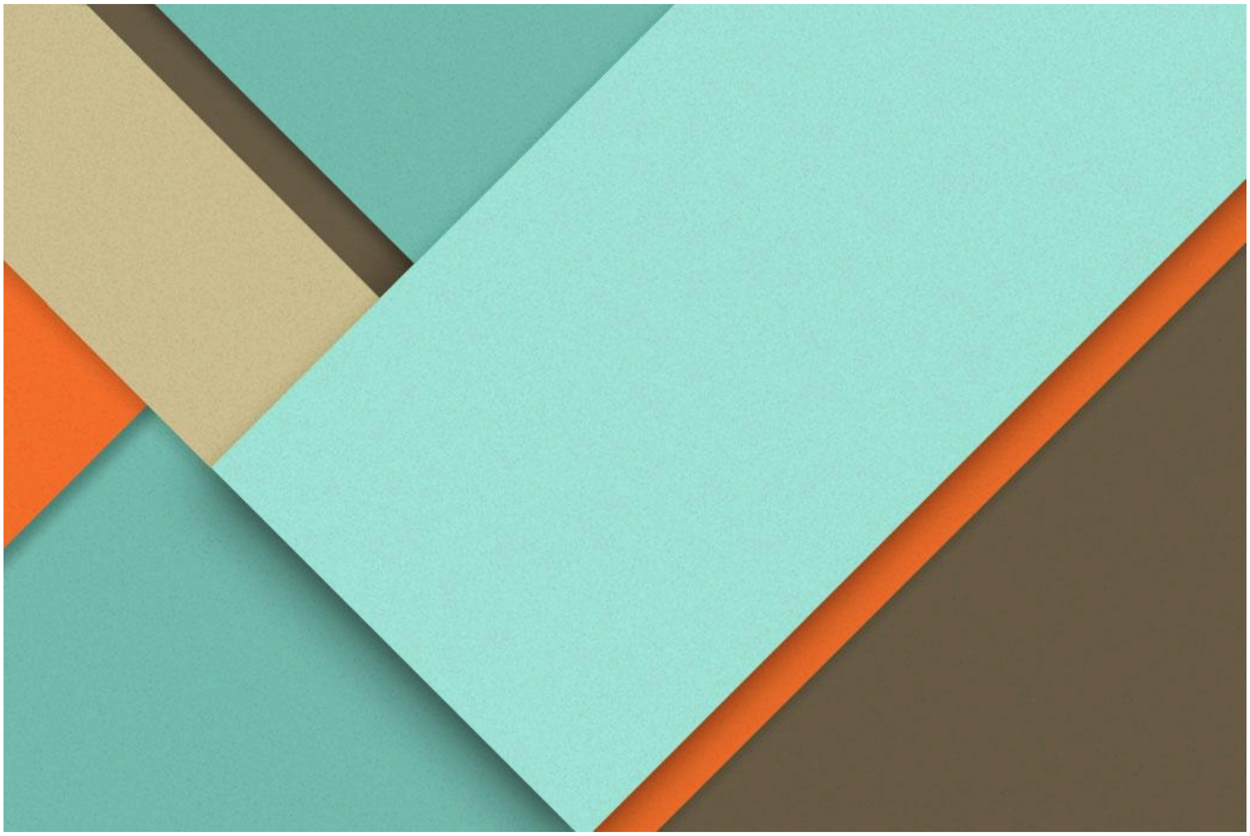


Cloud and Network Security-C1-2026

Week 4, Assignment 1

VLANs and Secure Switch Configuration

Jacob Matara

ID No: CS-CNS11-26059.

Sunday, February 15, 2026.

INTRODUCTION

This comprehensive lab focuses on the implementation and verification of multiple Layer 2 security features within a small enterprise network environment using Cisco Packet Tracer v.9.0. The physical and logical topology is centered around 2 switches which form the fabric and a router that serves as the network gateway and centralized DHCP server.

The topology consists of:

- One Cisco 4221 router (R1) acting as a DHCP server and default gateway.
- Two Cisco Catalyst 2960 switches (S1 & S2).
- Two PCs (PC-A and PC-B) both obtain IP addresses via DHCP.

The network uses VLAN 10 for both management and client access with R1 providing dynamic addressing services (***DHCP***) services on the ***192.168.10.0/24*** subnet to the connected endpoints, PC-A and PC-B.

The strategic focus of this lab was to move beyond default configurations to establish a hardened access layer. The key primary objectives were to establish secure ***802.1Q trunks*** using a non-default native VLAN to mitigate VLAN hopping risks; implement ***Port Security*** with different violation actions & aging types, ***"Sticky"*** MAC address learning and defined violation actions to prevent unauthorized device association and MAC flooding; deploying ***DHCP Snooping*** to defend against rogue DHCP servers & ***MITM*** (man in the middle) attacks and lastly applying ***PortFast*** & ***BPDU*** Guard to ensure rapid client convergence while protecting the ***Spanning Tree Protocol*** (STP) from unauthorized manipulation or loops.

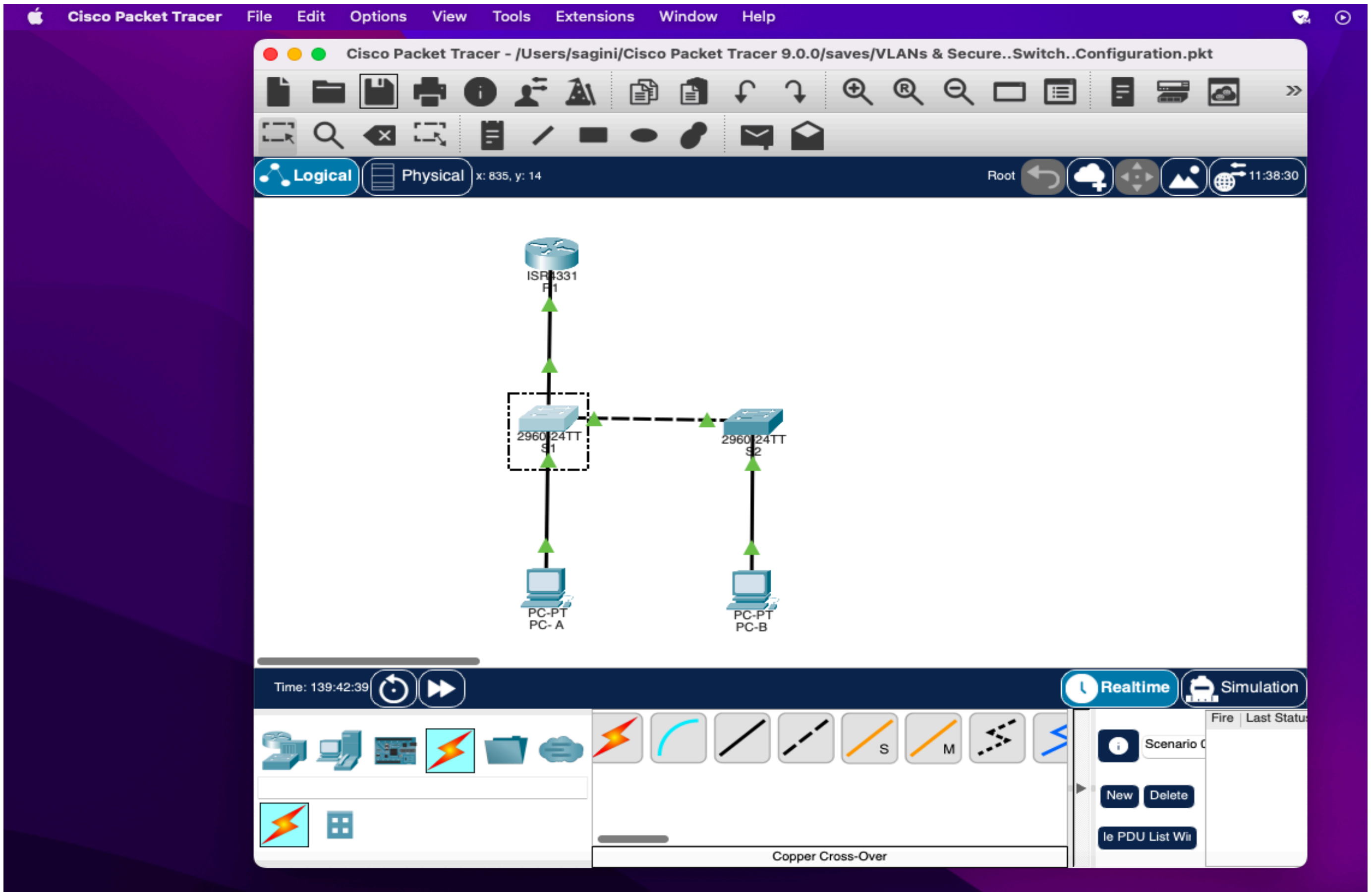
This lab enforces best practices for Layer 2 security through these configurations demonstrating a defense-in-depth approach to securing the local network infrastructure against common internal threat vectors..

Part 1: Network Device Initialization and Gateway Configuration

The initial phase of the lab focused on establishing the physical infrastructure and the logical foundation for automated network addressing. This stage ensured that the management plane and the control plane were ready for subsequent security hardening.

Physical Layer Connectivity

Following all the specified topology, the network was cabled to establish the backbone links between the gateway and the switching fabric. Key connections included the Router-to-Switch (R1 to S1) link via Gigabit Ethernet and the Inter-Switch Link (S1 to S2) via Fast Ethernet. This physical structure supports the later implementation of 802.1Q trunking.



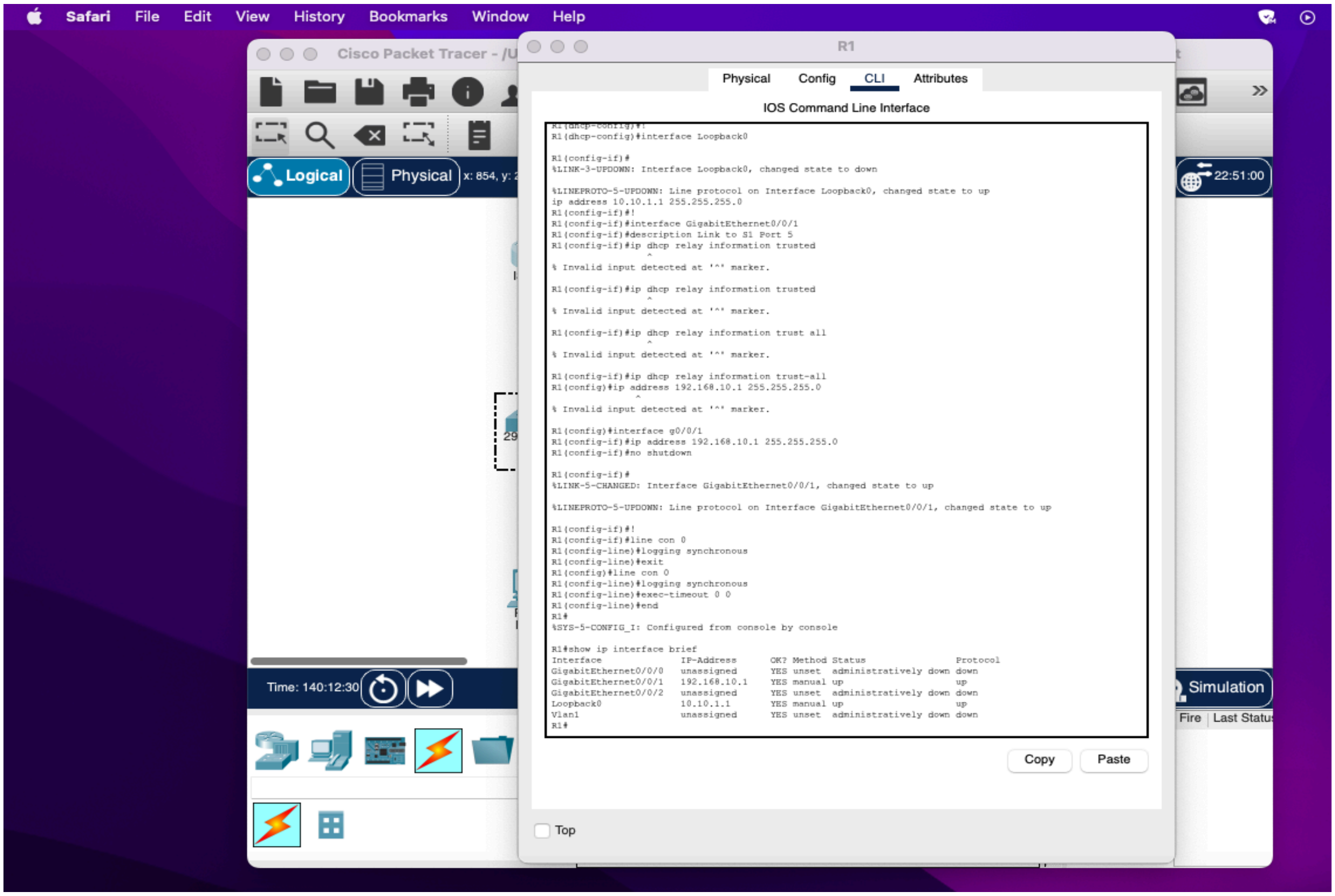
Router Configuration (R1 as a DHCP Server)

R1 was configured to act as the authoritative DHCP server for the local segment. The configuration focuses on address preservation and gateway availability.

- **DHCP Exclusion and Scoping:** IP ranges were excluded from the dynamic pool to protect static assignments for the gateway and management SVIs.
- **Interface Hardening:** The ***`ip dhcp relay information trusted`*** command was applied to G0/0/1 to ensure the router accepts DHCP messages with option 82 intel data which is critical once DHCP Snooping is enabled on the switches.

```
! Excluding addresses for Gateway and SVIs
ip dhcp excluded-address 192.168.10.1 192.168.10.9
ip dhcp excluded-address 192.168.10.201 192.168.10.202
! Defining the Student Pool
ip dhcp pool Students
 network 192.168.10.0 255.255.255.0
 default-router 192.168.10.1
 domain-name secure.com
```

The verification was positive as a successful initialization was confirmed using ***`show ip interface brief`*** verifying that both the physical gateway and the logical loopback interface are in an "*up/up*" status.



```
R1(dhcp-config)#!
R1(dhcp-config)#interface Loopback0

R1(config-if)#
%LINK-3-UPDOWN: Interface Loopback0, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
ip address 10.10.1.1 255.255.255.0
R1(config-if)#!
R1(config-if)#interface GigabitEthernet0/0/1
R1(config-if)#description Link to S1 Port 5
R1(config-if)#ip dhcp relay information trusted
                 ^
% Invalid input detected at '^' marker.

R1(config-if)#ip dhcp relay information trusted
                 ^
% Invalid input detected at '^' marker.

R1(config-if)#ip dhcp relay information trust all
                 ^
% Invalid input detected at '^' marker.

R1(config-if)#ip dhcp relay information trust-all
R1(config)#ip address 192.168.10.1 255.255.255.0
              ^
% Invalid input detected at '^' marker.

R1(config)#interface g0/0/1
R1(config-if)#ip address 192.168.10.1 255.255.255.0
R1(config-if)#no shutdown

R1(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/1, changed state to up

R1(config-if)#!
R1(config-if)#line con 0
R1(config-line)#logging synchronous
R1(config-line)#exit
R1(config)#line con 0
R1(config-line)#logging synchronous
R1(config-line)#exec-timeout 0 0
R1(config-line)#end
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0/0   unassigned      YES unset  administratively down down
GigabitEthernet0/0/1   192.168.10.1    YES manual up                    up
GigabitEthernet0/0/2   unassigned      YES unset  administratively down down
Loopback0              10.10.1.1       YES manual up                    up
Vlan1                  unassigned      YES unset  administratively down down
R1#
```

Baseline Switch Management

S1 and S2 were initialized with basic management parameters. This included defining hostnames and disabling DNS lookups to prevent CLI latency during command entry errors. Crucially, a default gateway was configured on both switches, enabling them to communicate with R1 across different network segments for more management.

Part 2: Logical Segmentation and VLAN Management

In part 2, the network was logically segmented into functional and security- oriented VLANs. This prevents the “flat network” vulnerability where a broadcast storm or an attack in one segment affects the entire infrastructure.

Management VLAN and SVI Implementation

VLAN 10 (*Management*) was created on both switches to isolate administrative traffic and client access. Switch Virtual Interfaces (SVIs) were then assigned to this VLAN, providing each switch with a unique IP address within the management subnet.

```
! S1 Management SVI Configuration
interface vlan 10
 description Management SVI
 ip address 192.168.10.201 255.255.255.0
 no shutdown
```

Security- Specific VLANs (Native & Parking_Lot)

To adhere to Layer 2 security best practices, two specialized VLANs were implemented;

- **VLAN 333 (Native):** Created to serve as a non-default VLAN for trunk links. By moving away from the default VLAN 1, the network is protected against “VLAN Hopping” and basic double-tagging attacks.
- **VLAN 999 (ParkingLot):** Designed as a dead-end VLAN. All unused ports will eventually be moved here and disabled, ensuring that an unauthorized physical connection does not provide immediate network access.

```
S1(config)# vlan 333
S1(config-vlan)# name Native
S1(config)# vlan 999
S1(config-vlan)# name ParkingLot
```

The `show vlan brief` was used to verify that the VLAN database on both switches correctly reflects these security segments before assigning them to physical interfaces.

Part 3: Configure Switch Security

While parts 1 and 2 established the network's foundation, part3 focused on transforming the infrastructure into a resilient environment capable of defending against common Layer 2 exploits. This phase implemented a "***Zero Trust***" approach to the access layer.

Hardening the Switching Fabric (Trunking & Access)

To prevent unauthorized data leakage between segments, the logical interfaces were strictly defined.

- **Static Trunking and DTP Mitigation:** Dynamic Trunking Protocol (DTP) was disabled using ***`switchport nonegotiate`*** to prevent "VLAN Hopping" via spoofed DTP packets. Furthermore, moving the native VLAN to VLAN 333 ensured that untagged traffic was isolated from the management data planes.
- **Infrastructure "ParkingLot":** All the unused ports were moved to VLAN 999 and administrative shutdown was applied. This ensured that a physical connection to an open wall jack did not grant a malicious actor an entry point into the network.

```
interface f0/1
 switchport mode trunk
 switchport trunk native vlan 333
 switchport nonegotiate
```

```
interface range f0/2-4 , f0/7-24 , g0/1-2
 switchport mode access
 switchport access vlan 999
 shutdown
```

Advanced Port Security Implementation

Port security was deployed on edge interfaces to mitigate MAC-Flooding attacks and prevent the connection of unauthorized hardware (like rogue access points).

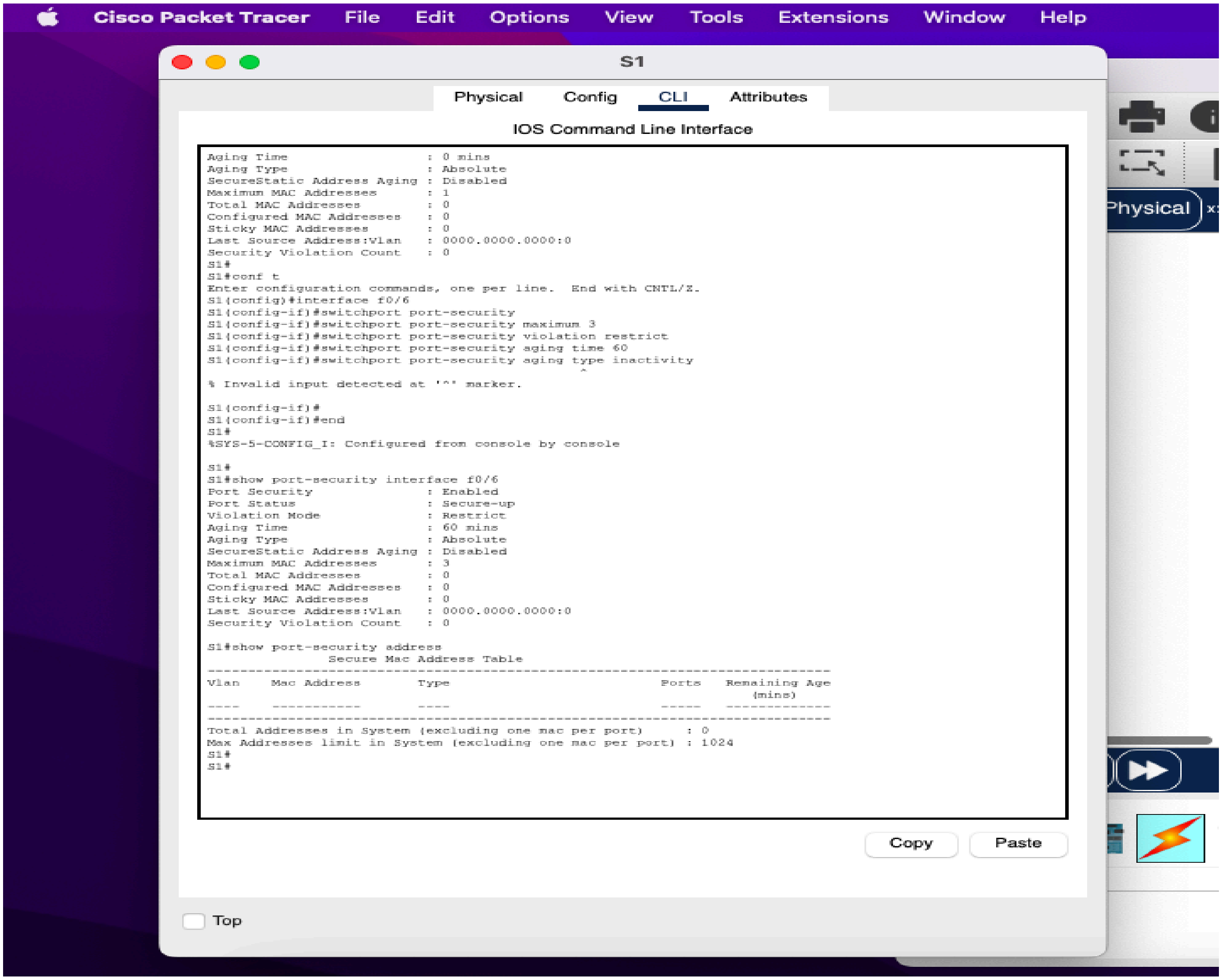
- **Adaptive Violations:** S1 was configured with the `restrict` action (dropping unauthorized packets and logging the event), while S2 utilized `protect` for a silent drop.
- **Sticky Learning:** By enabling `mac-address sticky`, the switches dynamically learn authorized MAC addresses and write them to the running configuration, providing persistent security across reboots without manual entry.

```
! Port Security Hardening on S1 (PC-A Interface)
interface f0/6
 switchport port-security
 switchport port-security maximum 3
 switchport port-security violation restrict
 switchport port-security aging time 60
 switchport port-security aging type inactivity
```

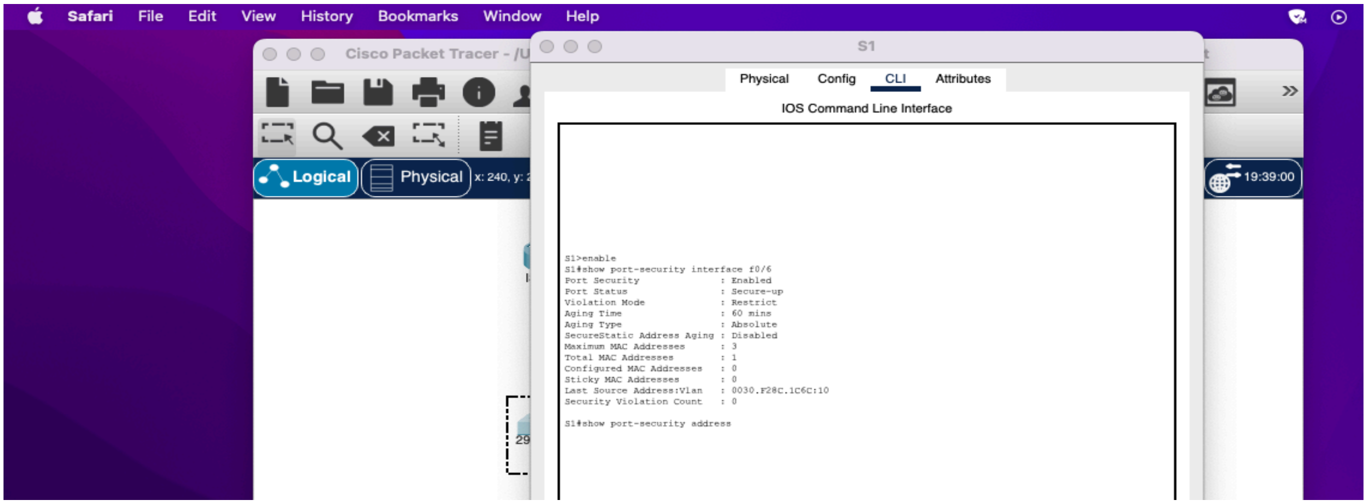
```
! Port Security Hardening on S2 (PC-A Interface)
interface f0/18
 switchport port-security
 switchport port-security mac-address sticky
 switchport port-security maximum 2
 switchport port-security violation protect
 switchport port-security aging time 60
```

The verification on both the switches was not quite a success as I utilized the `show port-security interface f0/6` & `show port-security interface f0/18` commands to obtain the addresses but at first I received a blank Last source address:Vlan 000 MAc address on both. Later on after all the configurations was done I enabled DHCP on PC-A which was on static and the address finally loaded 🎉.

The `show port-security interface address` is the only portion of the code that persisted and never changed because even after all configurations were complete, the address was still null/ blunk.



```
Aging Time                 : 0 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 1
Total MAC Addresses        : 0
Configured MAC Addresses   : 0
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0000.0000.0000:0
Security Violation Count   : 0
S1#
S1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
S1(config)#interface f0/6
S1(config-if)#switchport port-security
S1(config-if)#switchport port-security maximum 3
S1(config-if)#switchport port-security violation restrict
S1(config-if)#switchport port-security aging time 60
S1(config-if)#switchport port-security aging type inactivity
                                              ^
% Invalid input detected at '^' marker.

S1(config-if)#
S1(config-if)#end
S1#
%SYS-5-CONFIG_I: Configured from console by console

S1#
S1#show port-security interface f0/6
Port Security              : Enabled
Port Status                : Secure-up
Violation Mode             : Restrict
Aging Time                 : 60 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 3
Total MAC Addresses        : 0
Configured MAC Addresses   : 0
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0000.0000.0000:0
Security Violation Count   : 0

S1#show port-security address
               Secure Mac Address Table
----------------------------------------------------------------------------
Vlan    Mac Address       Type                          Ports   Remaining Age
                                                                   (mins)
----    -----------       ----                          -----   -------------
----------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 1024
S1#
S1#
```



```
S1>enable
S1#show port-security interface f0/6
Port Security              : Enabled
Port Status                : Secure-up
Violation Mode             : Restrict
Aging Time                 : 60 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 3
Total MAC Addresses        : 1
Configured MAC Addresses   : 0
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0030.F28C.1C6C:10
Security Violation Count   : 0

S1#show port-security address
```

Mitigating Man-in-the-Middle Attacks (DHCP Snooping)

To defend against “rogue DHCP Server” attacks, DHCP Snooping was implemented. This feature treats all access ports as “untrusted” by default.

- **Trust Anchors:** The uplink to R1 (via S1) was explicitly configured as a trusted port, allowing legitimate DHCP offers to reach clients.
- **Rate Limiting:** Access ports were limited to 5 DHCP packets per second to prevent DHCP starvation attacks which attempt to exhaust the router’s IP address pool.

```
! S2 DHCP Snooping Implementation
ip dhcp snooping
ip dhcp snooping vlan 10
interface f0/1
 ip dhcp snooping trust
interface f0/18
 ip dhcp snooping limit rate 5
```

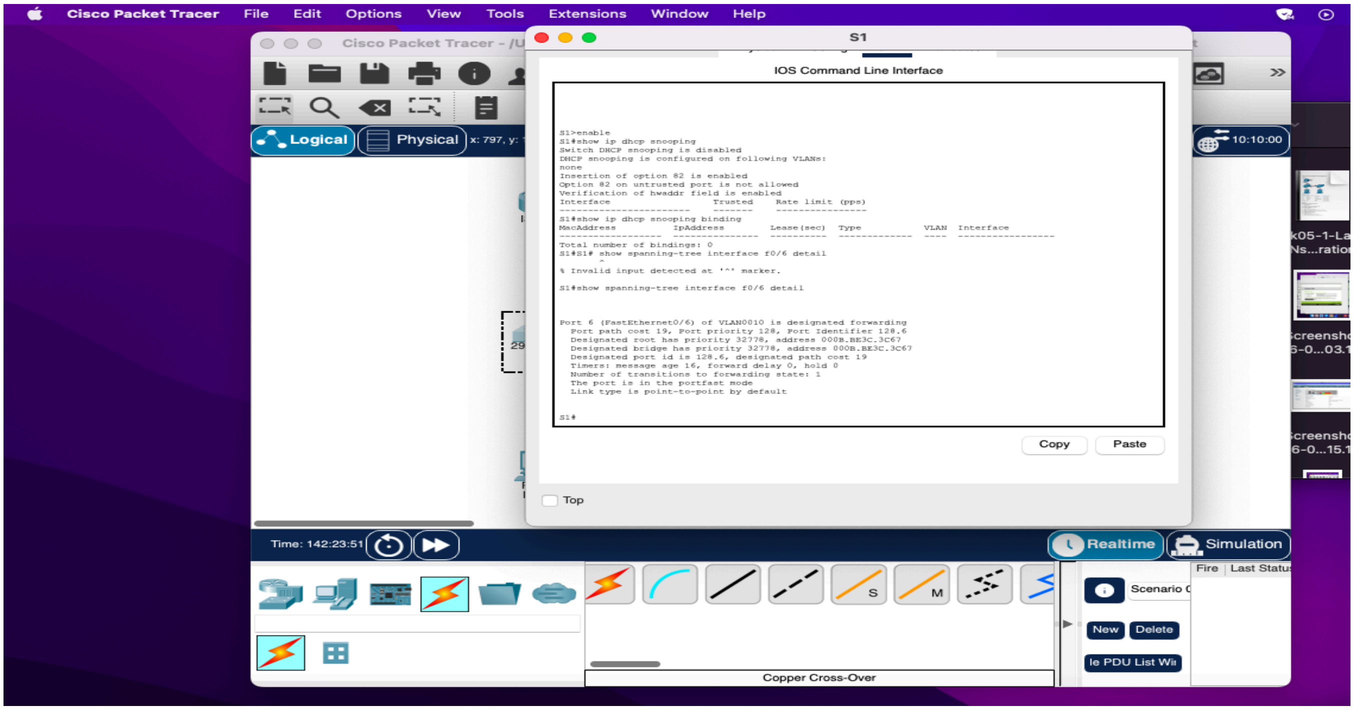
STP Stability with PortFast and BPDU Guard

To optimize performance and stability, Spanning Tree Protocol (STP) enhancements were applied to all edge ports.

- **PortFast:** Configured to allow PC-A and PC-B to bypass the listening/learning states, moving immediately to a forwarding state upon connection.
- **BPDU Guard:** A critical fail-safe that shuts down a PortFast- enabled port if it receives a Bridge Protocol Data Unit (BPDU). This prevents users from accidentally creating loops by connecting unauthorized switches to their desks.

```
! On S1
interface range f0/5 - 6
 spanning-tree portfast
 spanning-tree bpduguard enable
```

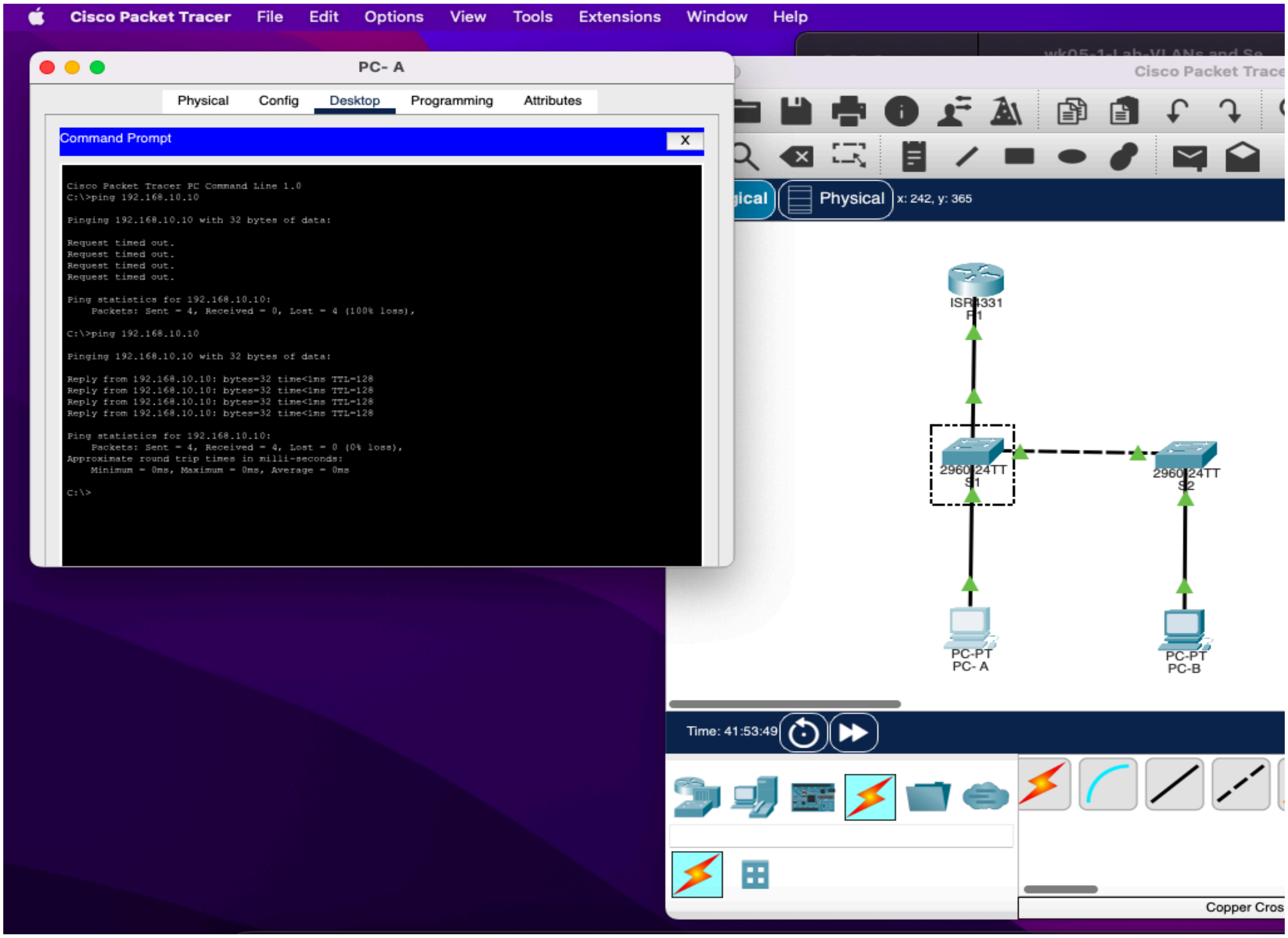
The output confirms that the interface is in PortFast mode and that BPDU Guard is enabled. If a BPDU is received on an access port, BPDU Guard will err-disable the port to protect the spanning-tree topology.



```
S1>enable
S1#show ip dhcp snooping
Switch DHCP snooping is disabled
DHCP snooping is configured on following VLANs:
none
Insertion of option 82 is enabled
Option 82 on untrusted port is not allowed
Verification of hwaddr field is enabled
Interface                    Trusted    Rate limit (pps)
-----------------------    -------    ----------------
S1#show ip dhcp snooping binding
MacAddress          IpAddress        Lease(sec)  Type           VLAN  Interface
------------------  ---------------  ----------  -------------  ----  -----------------
Total number of bindings: 0
S1#S1# show spanning-tree interface f0/6 detail
       ^
% Invalid input detected at '^' marker.

S1#show spanning-tree interface f0/6 detail



Port 6 (FastEthernet0/6) of VLAN0010 is designated forwarding
  Port path cost 19, Port priority 128, Port Identifier 128.6
  Designated root has priority 32778, address 000B.BE3C.3C67
  Designated bridge has priority 32778, address 000B.BE3C.3C67
  Designated port id is 128.6, designated path cost 19
  Timers: message age 16, forward delay 0, hold 0
  Number of transitions to forwarding state: 1
  The port is in the portfast mode
  Link type is point-to-point by default


S1#
```

Final Validation and Connectivity

The lab concluded with a comprehensive verification phase. Successful end-to-end communications were validated via ICMP (Ping) across all network tiers, including the simulated remote network (Loopback 0). This confirms that while the security posture is significantly hardened, the network remains fully operational and transparent for authorized users.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.10.10

Pinging 192.168.10.10 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.10.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>ping 192.168.10.10

Pinging 192.168.10.10 with 32 bytes of data:

Reply from 192.168.10.10: bytes=32 time<1ms TTL=128
Reply from 192.168.10.10: bytes=32 time<1ms TTL=128
Reply from 192.168.10.10: bytes=32 time<1ms TTL=128
Reply from 192.168.10.10: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.10.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

All connectivity tests I conducted passed, confirming that the Layer 2 security features did not prevent legitimate traffic when correctly configured.

Questions to Answer

1) **Why is there no timer value for the remaining age in minutes when sticky learning was configured on S2?**

   In the MacOS version of my pc, sticky secure MAC addresses are not aged out in the same way as dynamically learned addresses. Sticky MAC addresses are effectively “pinned” to the configuration (they are added to the running configuration) and are treated as static entries. As a result, the remaining age timer does not apply to sticky-secured MAC addresses, so no timer value is displayed.

2) **If you load the running-config script on S2, why will PC-B on port 18 never get an IP address via DHCP?**

   When the running configuration is reloaded, any sticky MAC addresses that were saved could already occupy the maximum number of allowed MAC addresses (2 in this case).violation mode “protect” silently drops excess frames, preventing DHCP packets from the legitimate PC-B MAC from reaching the server.Because the port is not shutdown and no violation counter incremental is obvious, PC-B appears connected but never receives a DHCP lease.

3) **What is the difference between absolute aging type and inactivity aging type?**

   In absolute aging, the secure MAC addresses are removed after the configured timer empires, regardless of whether traffic is seen or not. Once the timer runs out, all secure addresses on that port are aged out and must be relearned. Inactivity aging on the other hand ensures that each MAC address is removed only if no traffic is detected from that address for the duration of the timer. If a MAC address continues to send traffic, its inactivity timer is reset and the henry is retained.

CONCLUSION AND REFLECTIVE SUMMARY

This lab provided an extensive hands-on exploration of Layer 2 security mechanisms essential for hardening modern enterprise switching fabrics. Through the systematic implementation of security controls on Cisco Catalyst 2960 switches, I successfully transitioned a default, vulnerable environment into a resilient infrastructure.

The core technical takeaways from this exercise include:

- **Segmentation and Trunk Integrity:** By migrating to a non-default native VLAN (VLAN 333) and disabling DTP negotiation, I effectively neutralized common attack vectors such as VLAN hopping and double-tagging.
- **Surface Area Reduction:** The implementation of the “ParkingLot” (VLAN 999) and administrative shutdown of unused ports reinforced the principle of least privilege ensuring that physical access does not equate to network access.
- **Access Layer Intelligence:** Configuring port security with sticky learning and tailored violation modes (Restrict and Protect) demonstrated how to enforce device-level accountability. Furthermore, the deployment of DHCP snooping proved vital in maintaining the integrity of the address assignment process against rogue actors.
- **STP Resilience:** The integration of PortFast and BPDU Guard highlighted the critical balance between network performance (rapid convergence) and topology stability, protecting the Spanning Tree from accidental loops or malicious manipulation.

One of the most valuable aspects of this lab was the verification process of running the commands both on CLI and Cmd. That underscored a critical reality in network security where advanced features could silently impact legitimate connectivity if not meticulously configured. For instance, observing how a "***sticky***" MAC overflow in “*protect*” mode can drop traffic without administrative alerts highlighted the necessity of proactive monitoring and through CLI verification using `show` commands.

This lab reinforced the 🛡️Defense-in-Depth philosophy as no single feature, be it Port Security or DHCP Snooping is a silver bullet rather it is the layered application of these protocols that build a robust defense. This experience has deepened my understanding of switch-level forensics and configurations hardening, providing a 🪨 solid foundation for managing and securing enterprise-scale network deployments.

Theory is just the beginning but execution is the goal. Huge thanks to CyberShujaa for the hands-on training that turned technical protocols into professional triumphs! 🌐

Report compiled by [**Jacob Matara**]