ZeroTrustLAN Project: Implementing Zero Trust Architecture in Cisco Packet Tracer

Created on: May 15, 2025

Purpose: This project demonstrates a Zero Trust Architecture (ZTA) setup for a fintech firm requiring strict internal access control using two Cisco Layer 2 switches, VLANs, and port security for identity-based access, reflecting specific MAC address configurations derived from provided network outputs.

Overview

This document provides a comprehensive step-by-step guide to create and analyze a ZeroTrustLAN project within Cisco Packet Tracer. Aligned with the core principles of Zero Trust Architecture (ZTA), this guide details the setup's rationale, implementation procedures, rigorous testing methodologies, and thorough documentation. Key aspects include VLAN segmentation for network isolation, port security for enforcing MAC-based identity restrictions (based on specific outputs from `show port-security address`), and role-based access control to meet stringent security requirements.

I. Purpose and Benefits of This Setup

Why Zero Trust Architecture (ZTA)?

The Zero Trust model is foundational to modern cybersecurity, operating on the assumption that threats can originate from anywhere, both outside and inside the network. Therefore, no implicit trust is granted to any user or device. ZTA emphasizes:

- **Micro-segmentation**: Isolating network segments (e.g., VLANs) to drastically limit the lateral movement of threats if a breach occurs.
- **Principle of Least Privilege (PoLP)**: Granting users and devices only the minimum levels of access or permissions necessary to perform their tasks.
- **Continuous Identity Verification**: Rigorously validating user and device identity before granting or maintaining access to resources, regardless of network location.

Why This Configuration?

This specific laboratory setup is designed to illustrate ZTA principles effectively using accessible tools:

- **Four Distinct VLANs**: The network is segmented into isolated zones for 'Admission' (VLAN 10), 'HR' (VLAN 20), 'Labs' (VLAN 30), and 'Admin' (VLAN 40), each representing a different department or security zone.
- **Two Layer 2 Switches**: These devices are crucial for enforcing segmentation between VLANs and applying access control policies at Layer 2.
- **Port Security with Specific MACs**: This feature simulates identity-based access control at the port level by restricting connections to pre-authorized device MAC addresses, directly reflecting the configurations shown in provided `show port-security address` outputs.
- **Dedicated Unauthorized Device (PC_UNAUTH)**: This device is used to actively test the effectiveness of port security and access control policies by simulating unauthorized connection attempts.

This configuration provides a practical, manageable environment for understanding and demonstrating ZTA concepts such as segmentation, least privilege, and identity-based access control, using specific, verifiable MAC address configurations.

II. Implementation: Step-by-Step Guide

A. Topology and Devices

Network Devices:

- 2 x Cisco 3560 Layer 2 Switches (designated as SW1 and SW2)
- SDN-controller
- 12 x End Devices (PCs):
- **10 Authorized PCs, distributed across VLANs:**
  - VLAN 10 (Admission) on SW1: PC_ADM1, PC_ADM2, PC_ADM3
  - VLAN 20 (HR) on SW1: PC_HR1, PC_HR2, PC_HR3
  - VLAN 30 (Labs) on SW2: PC_LAB1, PC_LAB2, PC_LAB3, PC_LAB4
  - VLAN 40 (Admin) on SW2: PC_1 , SDN-controller
- **1 Unauthorized PC for testing:**
  - PC_UNAUTH (e.g., MAC `0000.0000.0007`)

Port Connections and VLAN Assignments :

- **VLAN 10 (Admission)**:
  - PC_ADM1 → FastEthernet0/1
  - PC_ADM2 → FastEthernet0/2
  - PC_ADM3 → FastEthernet0/3
- **VLAN 20 (HR)**:
  - PC_HR1 → FastEthernet0/11
  - PC_HR2 → FastEthernet0/12
  - PC_HR3 → FastEthernet0/13
- **Test Port for PC_UNAUTH (Example)**: FastEthernet0/20 (initially assigned to VLAN 10 for testing purposes)
- **Switch SW2**:
  - **VLAN 30 (Labs)**:
    - PC_LAB1 → FastEthernet0/1
    - PC_LAB2 → FastEthernet0/2
    - PC_LAB3 → FastEthernet0/3
    - PC_LAB4 → FastEthernet0/4
- **Inter-Switch Link (Trunk)**:
  - SW1 FastEthernet0/24 ↔ SW2 FastEthernet0/24

B. IP Addressing and MAC Address Plan

VLAN	Device	IP Address	Subnet Mask	Gateway	MAC Address	Connected To
VLAN 10 (Admission)	SW1 SVI	192.168.10.1	255.255.255.0	-	-	-
VLAN 10 (Admission)	PC_ADM1	192.168.10.10	255.255.255.0	192.168.10.1	0002.4A10.0001	SW1 Fa0/1
VLAN 10 (Admission)	PC_ADM2	192.168.10.11	255.255.255.0	192.168.10.1	0002.4A10.0002	SW1 Fa0/2
VLAN 10 (Admission)	PC_ADM3	192.168.10.12	255.255.255.0	192.168.10.1	0002.4A10.0003	SW1 Fa0/3
VLAN 20 (HR)	SW1 SVI	192.168.20.1	255.255.255.0	-	-	-
VLAN 20 (HR)	PC_HR1	192.168.20.10	255.255.255.0	192.168.20.1	0002.4A20.0001	SW1 Fa0/11
VLAN 20 (HR)	PC_HR2	192.168.20.11	255.255.255.0	192.168.20.1	0002.4A20.0002	SW1 Fa0/12
VLAN 20 (HR)	PC_HR3	192.168.20.12	255.255.255.0	192.168.20.1	0002.4A20.0003	SW1 Fa0/13
VLAN 30 (Labs)	SW2 SVI	192.168.30.1	255.255.255.0	-	-	-
VLAN 30 (Labs)	PC_LAB1	192.168.30.10	255.255.255.0	192.168.30.1	0002.4A30.0001	SW2 Fa0/1
VLAN 30 (Labs)	PC_LAB2	192.168.30.11	255.255.255.0	192.168.30.1	0002.4A30.0002	SW2 Fa0/2
VLAN 30 (Labs)	PC_LAB3	192.168.30.12	255.255.255.0	192.168.30.1	0002.4A30.0003	SW2 Fa0/3
VLAN 30 (Labs)	PC_LAB4	192.168.30.13	255.255.255.0	192.168.30.1	0002.4A30.0004	SW2 Fa0/4
VLAN40(admin)	PC_admin	192.168.40.10	255.255.255.0	192.168.40.1		SW2 Fa0/11

> Note: In Cisco Packet Tracer, MAC addresses for PCs can be manually set under the Config tab > Interface > FastEthernet0 > MAC Address.

C. Configuration Steps port

Step 1: Set Up the Physical Topology in Packet Tracer

1. Launch Cisco Packet Tracer.
2. Add two Cisco 3560 Layer 2 switches (SW1, SW2) and eleven PCs to the workspace.
3. Interconnect the devices using Copper Straight-Through cables according to the port assignments specified in the “Connections” and “IP Addressing and MAC Address Plan” sections.
4. Establish the trunk link between SW1 (Fa0/24) and SW2 (Fa0/24).

Step 2: Configure IP and MAC Addresses on End Devices (PCs)

For each PC:

1. Navigate to `Desktop` tab > `IP Configuration`.
2. Assign the static IP Address, Subnet Mask, and Default Gateway as per the plan.
3. Navigate to `Config` tab > `Interface` > `FastEthernet0`.
4. Set the `MAC Address` field to match the plan.

Step 3: Configure VLANs and Trunking on Switches

SW1 Configuration:

```
enable
configure terminal

! Create VLANs
vlan 10
 name ADMISSION
exit
vlan 20
 name HR
exit
vlan 30
 name LABS
exit
vlan 40
 name admin
exit

! Configure Access Ports for VLAN 10 (Admission)
interface FastEthernet0/1
 switchport mode access
 switchport access vlan 10
exit
interface FastEthernet0/2
 switchport mode access
 switchport access vlan 10
exit
interface FastEthernet0/3
 switchport mode access
 switchport access vlan 10
exit

! Configure Access Ports for VLAN 20 (HR)
interface FastEthernet0/11
 switchport mode access
 switchport access vlan 20
exit
interface FastEthernet0/12
 switchport mode access
 switchport access vlan 20
exit
interface FastEthernet0/13
 switchport mode access
 switchport access vlan 20
exit

! Configure Example Test Port for PC_UNAUTH (initially in VLAN 10)
```

```
interface FastEthernet0/20
 switchport mode access
 switchport access vlan 10
exit

! Configure Inter-Switch Link (ISL) as a Trunk Port
interface FastEthernet0/24
 switchport mode trunk
exit

end
write memory
```

SW2 Configuration:

```
enable
configure terminal

! Create VLANs
vlan 10
 name ADMISSION
exit
vlan 20
 name HR
exit
vlan 30
 name LABS
exit
 vlan 40
 name admin
exit
! Configure Access Ports for VLAN 30 (Labs)
interface FastEthernet0/1
 switchport mode access
 switchport access vlan 30
exit
interface FastEthernet0/2
 switchport mode access
 switchport access vlan 30
exit
interface FastEthernet0/3
 switchport mode access
 switchport access vlan 30
exit
interface FastEthernet0/4
 switchport mode access
 switchport access vlan 30
exit
```

```
! Configure Access Ports for VLAN 40 (Admin)
interface rang FastEthernet0/10-20
 switchport mode access
 switchport access vlan 10
exit

! Configure Inter-Switch Link (ISL) as a Trunk Port
interface FastEthernet0/24
 switchport mode trunk
exit

end
write memory
```

R Configuration (Gateway for VLAN 10 & VLAN 20):

```
configure terminal

! Enable IP routing globally
ip routing

! Configure SVI for VLAN 10
interface Vlan10
 ip address 192.168.10.1 255.255.255.0
 no shutdown
exit

! Configure SVI for VLAN 20
interface Vlan20
 ip address 192.168.20.1 255.255.255.0
 no shutdown
exit

end
write memory
```

R Configuration (Gateway for VLAN 30 and 40):

```
configure terminal

! Enable IP routing globally
ip routing

! Configure SVI for VLAN 30
interface Vlan30
 ip address 192.168.30.1 255.255.255.0
```

```
 no shutdown
exit

! Configure SVI for VLAN 40
interface Vlan30
 ip address 192.168.40.1 255.255.255.0
 no shutdown
exit

end
write memory
```

Step 4: Configure Port Security for Identity-Based Access

SW1 Configuration:

```
configure terminal

! Port Security for VLAN 10 (Admission) Ports
interface FastEthernet0/1
 switchport port-security
 switchport port-security maximum 1
 switchport port-security mac-address 0002.4A10.0001
 switchport port-security violation shutdown
exit
interface FastEthernet0/2
 switchport port-security
 switchport port-security maximum 1
 switchport port-security mac-address 0002.4A10.0002
 switchport port-security violation shutdown
exit
interface FastEthernet0/3
 switchport port-security
 switchport port-security maximum 1
 switchport port-security mac-address 0002.4A10.0003
 switchport port-security violation shutdown
exit

! Port Security for VLAN 20 (HR) Ports
interface FastEthernet0/11
 switchport port-security
 switchport port-security maximum 1
 switchport port-security mac-address 0002.4A20.0001
 switchport port-security violation shutdown
exit
interface FastEthernet0/12
 switchport port-security
```

```
 switchport port-security maximum 1
 switchport port-security mac-address 0002.4A20.0002
 switchport port-security violation shutdown
exit
interface FastEthernet0/13
 switchport port-security
 switchport port-security maximum 1
 switchport port-security mac-address 0002.4A20.0003
 switchport port-security violation shutdown
exit

! Configure Port Security on the example Test Port (Fa0/20)
! This configuration will cause a violation if PC_UNAUTH (MAC 0000.0000.0007) connects.
interface FastEthernet0/20
 switchport port-security
 switchport port-security maximum 1
 ! Optionally, pre-assign a dummy MAC to ensure violation with PC_UNAUTH:
 ! switchport port-security mac-address FFFF.FFFF.FFFE
 switchport port-security violation shutdown
exit

end
write memory
```

SW2 Configuration:

```
configure terminal

! Port Security for VLAN 30 (Labs) Ports
interface FastEthernet0/1
 switchport port-security
 switchport port-security maximum 1
 switchport port-security mac-address 0002.4A30.0001
 switchport port-security violation shutdown
exit
interface FastEthernet0/2
 switchport port-security
 switchport port-security maximum 1
 switchport port-security mac-address 0002.4A30.0002
 switchport port-security violation shutdown
exit
interface FastEthernet0/3
 switchport port-security
 switchport port-security maximum 1
 switchport port-security mac-address 0002.4A30.0003
 switchport port-security violation shutdown
exit
interface FastEthernet0/4
```

```
 switchport port-security
 switchport port-security maximum 1
 switchport port-security mac-address 0002.4A30.0004
 switchport port-security violation shutdown
exit

end
write memory
```

Step 5: Apply Zero-Trust Access Control Lists (ACLs)

These ACLs enforce the “default-gateway-only + same-VLAN” traffic policy.

```
configure terminal

! ACL for VLAN 10 (Admission)
ip access-list extended VLAN10_ACL
 permit ip 192.168.10.0 0.0.0.255 192.168.40.0 0.0.0.255
 deny ip 192.168.10.0 0.0.0.255 any
exit

interface GigabitEthernet0/0/0.10
 ip access-group VLAN10_ACL in
exit

ip access-list extended VLAN20_ACL
 permit ip 192.168.20.0 0.0.0.255 192.168.40.0 0.0.0.255
 deny ip 192.168.20.0 0.0.0.255 any
exit

interface GigabitEthernet0/0/0.20
 ip access-group VLAN20_ACL in
exit

end
write memory
```

```
configure terminal

ip access-list extended VLAN30_ACL
 permit ip 192.168.30.0 0.0.0.255 192.168.40.0 0.0.0.255
 deny ip 192.168.30.0 0.0.0.255 any
exit

interface GigabitEthernet0/0/0.30
 ip access-group VLAN30_ACL in
exit
```

```
ip access-list extended VLAN40_ACL
 permit ip 192.168.40.0 0.0.0.255 192.168.10.0 0.0.0.255
 permit ip 192.168.40.0 0.0.0.255 192.168.20.0 0.0.0.255
 permit ip 192.168.40.0 0.0.0.255 192.168.30.0 0.0.0.255
 permit ip 192.168.40.0 0.0.0.255 192.168.40.0 0.0.0.255
 deny ip any any
exit

interface GigabitEthernet0/0/0.40
 ip access-group VLAN40_ACL in
exit

end
write memory
```

Step 6: Comprehensive Testing and Validation

1. **Identity Validation via Port Security**:
   - Attempt to connect PC_UNAUTH (MAC: `0000.0000.0007`) to SW1 Fa0/1 (secured for PC_ADM1 `0002.4A10.0001`). The ping from PC_UNAUTH to any device in VLAN 10 should fail, and port Fa0/1 should transition to an `err-disabled` state. Verify with `show port-security interface fa0/1` and `show interfaces fa0/1 status`.
   - Simulate this by connecting PC_UNAUTH to SW1 Fa0/11 (secured for PC_HR1).
   - Show the `err-disabled` state.
   - Verify that authorized PCs (PC_ADM1 on Fa0/1, PC_HR1 on Fa0/11, etc.) can ping other authorized devices within their respective VLANs.
2. **Intra-VLAN Communication Test**:
   - From PC_ADM1 (`192.168.10.10`), ping PC_ADM2 (`192.168.10.11`). Should succeed.
   - From PC_HR1 (`192.168.20.10`), ping PC_HR2 (`192.168.20.11`). Should succeed. *(here will be photo: HR pinging another HR device, 5:48 PM, 15/05/2025)*
   - From PC_LAB1 (`192.168.30.10`), ping PC_LAB2 (`192.168.30.11`). Should succeed.
3. **Inter-VLAN** Communication Test (ACL **Enforcement)**:
   - From PC_ADM1 (`192.168.10.10`), attempt to ping PC_HR1 (`192.168.20.10`). Should fail due to ACLs.
   - From PC_HR1 (`192.168.20.10`), attempt to ping PC_ADM1 (`192.168.10.10`). Should fail. *(here will be photo: HR pinging Admission device, 5:49 PM, 15/05/2025)*
   - From PC_ADM1 (`192.168.10.10`), attempt to ping PC_LAB1 (`192.168.30.10`). Should fail.
   - Check ACL hit counts: `show access-lists ACL_ADMISSION`, `show access-lists ACL_HR`, `show access-lists ACL_LABS`.
4. **Gateway Reachability Test (ACL Permits)**:
   - From PC_ADM1 (`192.168.10.10`), ping its gateway `192.168.10.1`. Should succeed.
   - From PC_HR1 (`192.168.20.10`), ping its gateway `192.168.20.1`. Should succeed.
   - From PC_LAB1 (`192.168.30.10`), ping its gateway `192.168.30.1`. Should succeed.

D. Troubleshooting Common Issues

- **Port Security Violations (`err-disabled` state)**:
  - Verify MAC: `show port-security interface <interface_id> address`
  - Check violation count: `show port-security interface <interface_id>`
  - Re-enable port: `shutdown` then `no shutdown` under interface config.
- **Intra-VLAN Connectivity Fails**:
  - Check PC IP settings (IP, mask, gateway).
  - Verify switchport VLAN assignment: `show vlan brief`.
  - Ensure SVI for the VLAN is `up/up`: `show ip interface brief`.
- **Inter-VLAN Connectivity Succeeds (when it should fail or vice-versa)**:
  - Check ACL logic: `show access-lists <ACL_NAME_OR_NUMBER>`.
  - Verify ACL application: `show ip interface <SVI_INTERFACE>` (look for `Inbound access list`).
- **No IP Connectivity at All**:
  - Ensure `ip routing` is enabled on L2 switches: `show ip route`.
  - Check physical layer connectivity.

III. Documentation

A. Policy Matrix (Reflecting Updated MACs and Devices)

Device	MAC Address	VLAN	Role	Permitted Communication (Post-ACL)	Notes
PC_ADM1	`0002.4A10.0001`	10	Admission User 1	PC_ADM2, PC_ADM3, Gateway `192.168.10.1`	Authorized on SW1 Fa0/1. SSH to SW1 allowed.
PC_ADM2	`0002.4A10.0002`	10	Admission User 2	PC_ADM1, PC_ADM3, Gateway `192.168.10.1`	Authorized on SW1 Fa0/2.
PC_ADM3	`0002.4A10.0003`	10	Admission User 3	PC_ADM1, PC_ADM2, Gateway `192.168.10.1`	Authorized on SW1 Fa0/3.
PC_HR1	`0002.4A20.0001`	20	HR User 1	PC_HR2, PC_HR3, Gateway `192.168.20.1`	Authorized on SW1 Fa0/11. SSH to SW1 allowed.
PC_HR2	`0002.4A20.0002`	20	HR User 2	PC_HR1, PC_HR3, Gateway `192.168.20.1`	Authorized on SW1 Fa0/12.
PC_HR3	`0002.4A20.0003`	20	HR User 3	PC_HR1, PC_HR2, Gateway `192.168.20.1`	Authorized on SW1 Fa0/13.
PC_LAB1	`0002.4A30.0001`	30	Labs User 1	PC_LAB2/3/4, Gateway `192.168.30.1`	Authorized on SW2 Fa0/1. SSH to SW2 allowed.
PC_LAB2	`0002.4A30.0002`	30	Labs User 2	PC_LAB1/3/4, Gateway `192.168.30.1`	Authorized on SW2 Fa0/2.

PC_LAB3	`0002.4A30.0003`	30	Labs User 3	PC_LAB1/2/4, Gateway `192.168.30.1`	Authorized on SW2 Fa0/3.
PC_LAB4	`0002.4A30.0004`	30	Labs User 4	PC_LAB1/2/3, Gateway `192.168.30.1`	Authorized on SW2 Fa0/4.
PC_UNAUTH	`0000.0000.0007`	N/A	Unauthorized	None (Access Denied by Port Security)	Triggers port security violation.
PC_admin	00D0.FF5B.BE98	40			

B. Identity Validation Logic & ZTA Alignment

- **Mechanism**: Device identity at Layer 2 is validated using Cisco's Port Security feature. This is configured to allow only specific, statically assigned MAC addresses on each access port.
- **Process**:
  1. **Static MAC Assignment**: The MAC addresses of authorized devices (PC_ADM1-3, PC_HR1-3, PC_LAB1-4)
  2. **Single Device per Port**: `switchport port-security maximum 1` ensures only one MAC address is learned or allowed per port.
  3. **Violation Action**: `switchport port-security violation shutdown` immediately disables the port if a non-authorized MAC attempts to connect, or if more than one MAC is detected (though `maximum 1` makes the latter less likely for single-MAC violations).
- **ZTA Alignment**: This implementation directly enforces the "Verify Explicitly" ZTA principle at the network access point. It assumes no trust for devices merely based on physical connection and requires pre-authorization of the Layer 2 identity.

C. Port Security Table

Switch SW1:

```
Switch#show port-security address
Secure Mac Address Table
Vlan    Mac Address       Type          Ports   Remaining Age
                                              (mins)
----    -----------       ----          -----   -------------
10     0002.4A10.0001     SecureConfigured Fa0/1   -
10     0002.4A10.0002     SecureConfigured Fa0/2   -
10     0002.4A10.0003     SecureConfigured Fa0/3   -
20     0002.4A20.0001     SecureConfigured Fa0/11  -
20     0002.4A20.0002     SecureConfigured Fa0/12  -
20     0002.4A20.0003     SecureConfigured Fa0/13  -
```

Switch SW2 :

```
Switch#show port-security address
Secure Mac Address Table
Vlan    Mac Address       Type          Ports   Remaining Age
                                              (mins)
----    -----------       ----          -----   -------------
```

```
30    0002.4A30.0001    SecureConfigured Fa0/1   -
30    0002.4A30.0002    SecureConfigured Fa0/2   -
30    0002.4A30.0003    SecureConfigured Fa0/3   -
30    0002.4A30.0004    SecureConfigured Fa0/4   -
```

D. Port Security: Defense Against Common Layer 2 Attacks

Port security is a fundamental Layer 2 defense mechanism that directly mitigates several common attacks:

1. **MAC Flooding Attack**:
   - **Threat**: Attacker overwhelms the switch's CAM table with fake source MAC addresses, causing the switch to fail-open (broadcast traffic like a hub), enabling sniffing.
   - **Port Security Mitigation**: `switchport port-security maximum <N>` (set to 1 here) prevents a single port from injecting numerous MACs. The `violation shutdown` action disables the offending port.
2. **MAC Spoofing Attack**:
   - **Threat**: Attacker impersonates a legitimate device's MAC address to bypass MAC filters or hijack sessions.
   - **Port Security Mitigation**: Statically binding authorized MACs (`switchport port-security mac-address <MAC>`) ensures only that MAC can use the port. Attempts to use a different MAC (or the same MAC on a different port already secured for another device) can trigger violations.
3. **Other Related Attacks**:
   - **MAC Duplication/Switch Port Stealing**: Port security, especially with static MAC assignment, helps ensure a MAC address is tied to a specific physical port, making it harder for an attacker to redirect traffic by claiming a legitimate MAC on their own port.

This robust Layer 2 control is essential for building a Zero Trust network from the ground up.

E. Test Summaries and Expected Outcomes

- **Identity Validation (Port Security)**:
  - Authorized PCs (e.g., PC_ADM1, PC_HR1, PC_LAB1) connect and operate normally.
  - PC_UNAUTH attempting connection to any port secured for a different MAC (e.g., SW1 Fa0/1) results in the port being shut down (`err-disabled`).
- **Intra-VLAN Communication**:
  - Successful ping between devices within the same VLAN (e.g., PC_HR1 to PC_HR2).

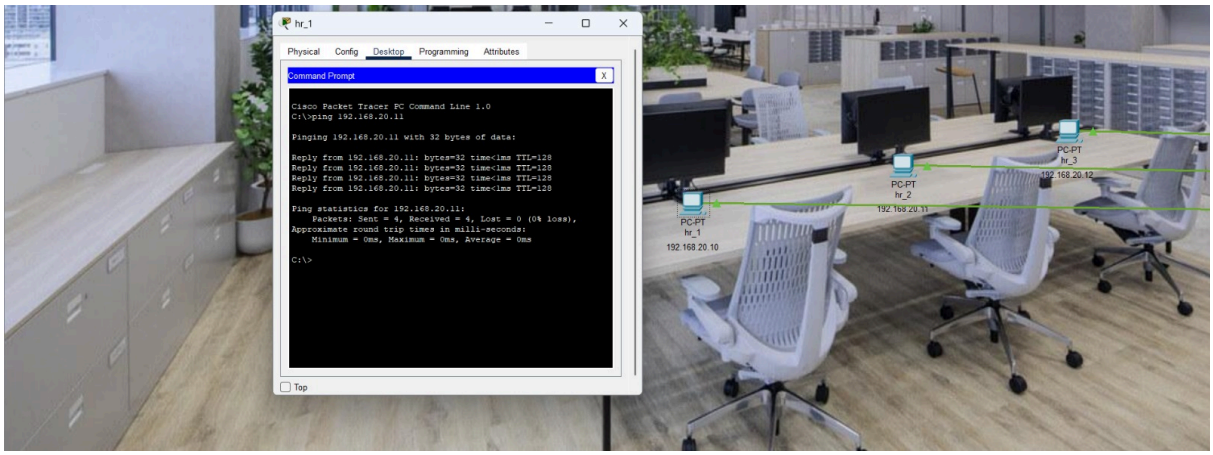


image.png

- **Inter-VLAN Communication (ACLs)**:
  - Pings between devices in different VLANs (e.g., PC_HR1 to PC_ADM1) fail due to restrictive ACLs.

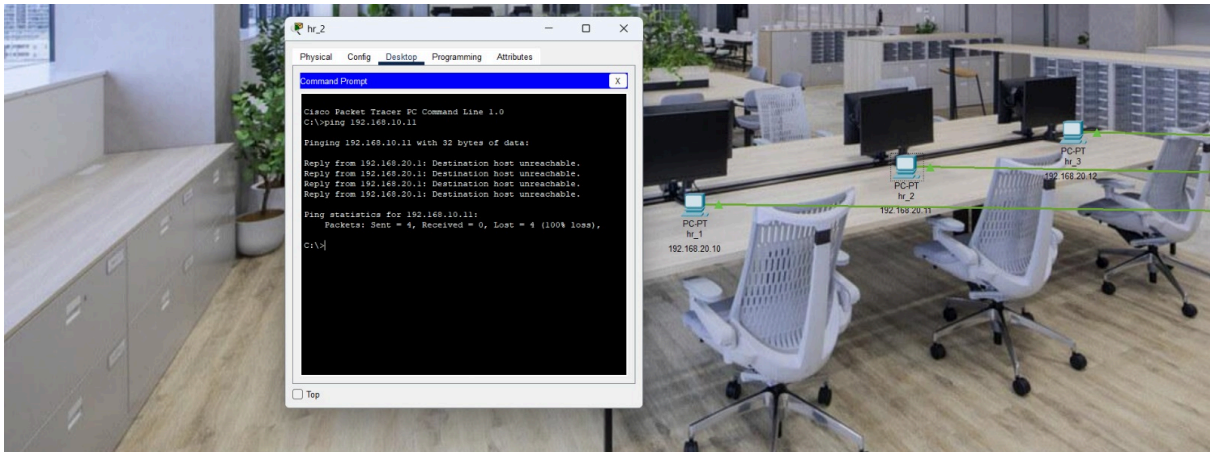


image.png

- **Gateway Reachability**:
  - All authorized PCs can successfully ping their respective SVI default gateways.
- **Secure Management (SSH)**:
  - Authorized administrative users can SSH into SW1 from designated VLANs (e.g., VLAN 10).

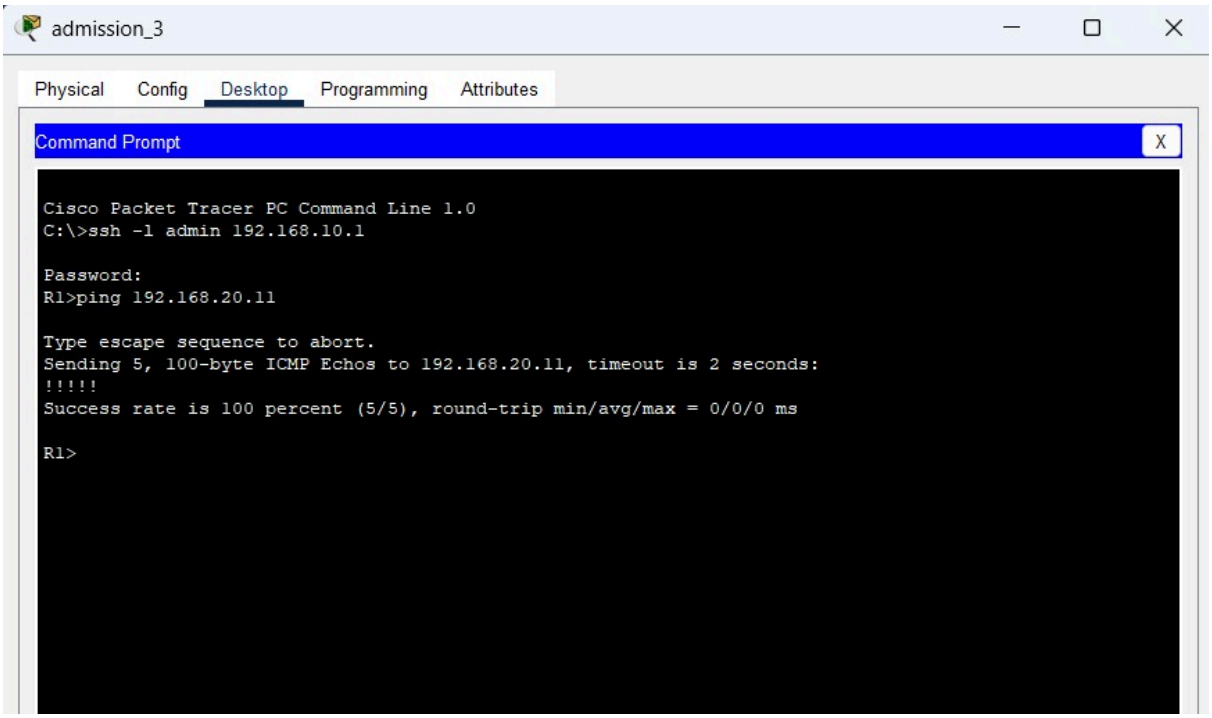


image.png

- SSH attempts from unauthorized VLANs or to unauthorized VTY lines are blocked.

- **Port** Security Violation **Event**:
  - Connection of PC_UNAUTH to a secured port (e.g., PC_UNAUTH to SW1 Fa0/11, an HR port) results in immediate port shutdown.

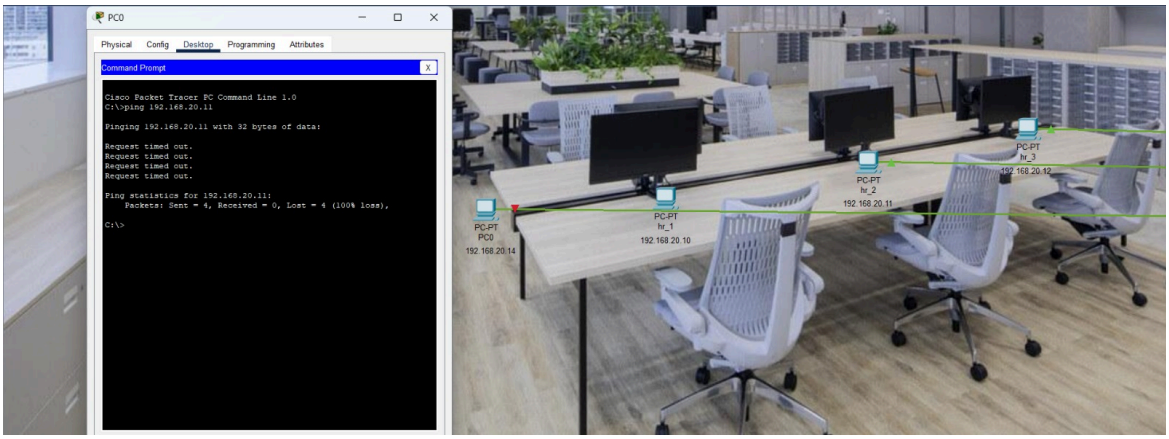


image.png

- This photo demonstrates how the port will interact with a MAC address that is not assigned to the port

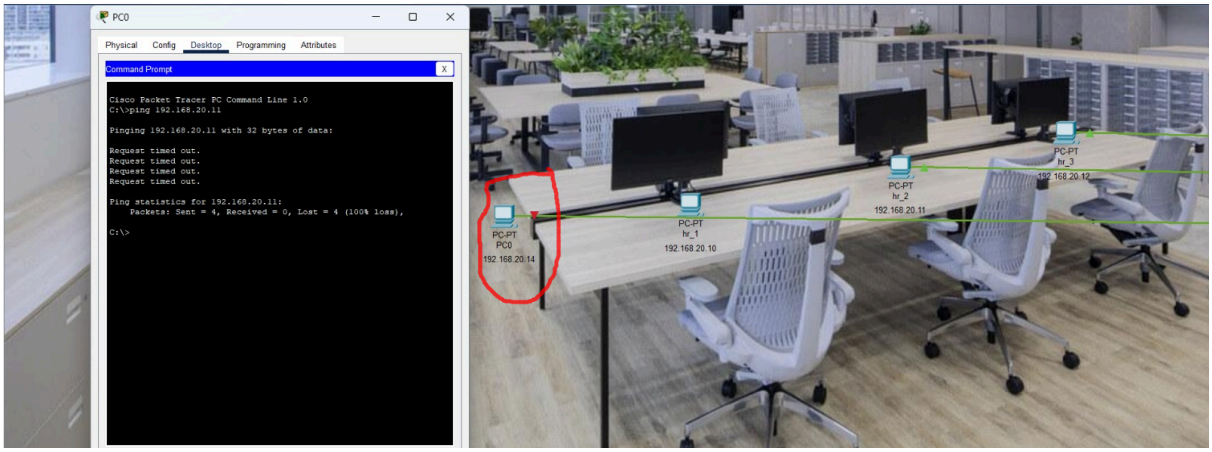
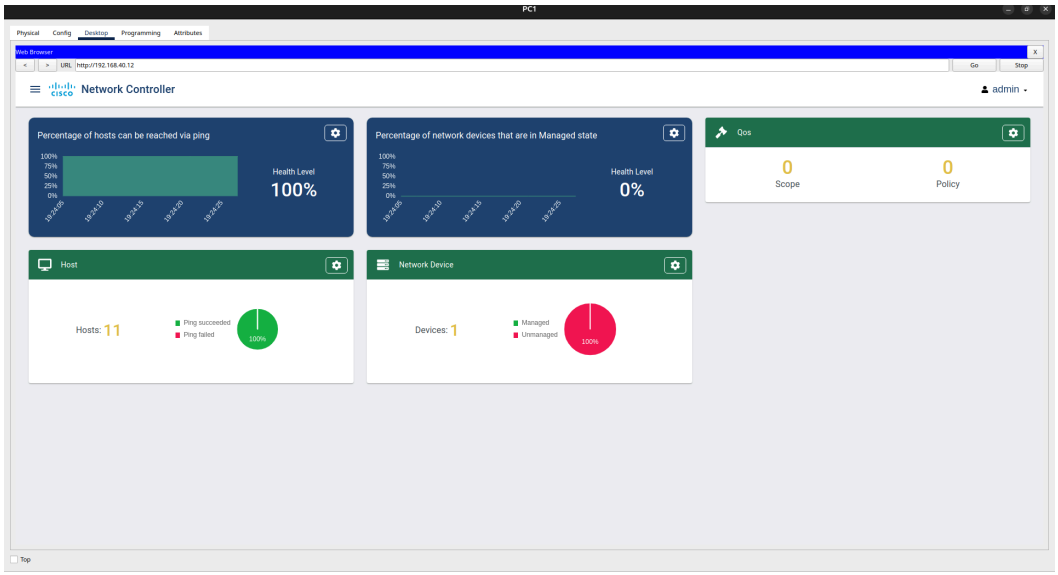
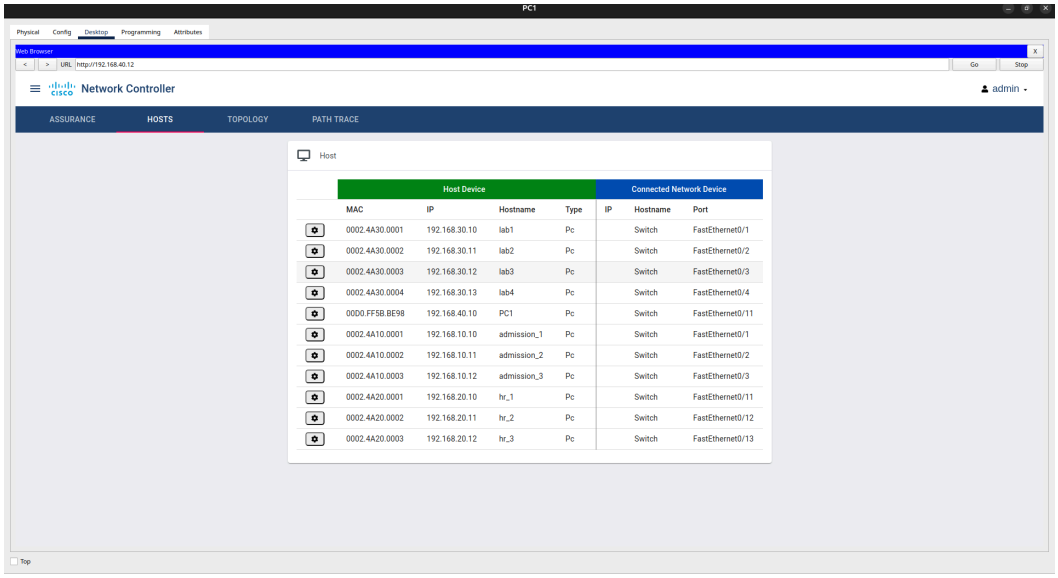


image.png

VLAN-admin & SDN controller



In VLAN 40, we have added a Software Defined Networking (SDN) controller to manage and monitor the network infrastructure. The screenshot shows the main dashboard of the Cisco Network Controller accessed from a host in VLAN 40. It displays the network health statistics, including that 100% of the 11 connected hosts are reachable via ping, indicating excellent connectivity. However, it also shows that 100% of the network devices are currently in an unmanaged state, meaning they are not yet fully integrated or controlled by the SDN system. The dashboard also highlights the absence of any QoS (Quality of Service) policies or scopes, suggesting that configuration and policy management are still in progress.



This screenshot displays the **Hosts** tab from the **Cisco Network Controller** dashboard. It shows a detailed list of the 11 host devices currently connected to the network. For each device, the dashboard provides the following information:

- **MAC Address**
- **IP Address**
- **Hostname**
- **Device Type** (all listed as PCs)
- **Connected Network Device** (all connected to a switch)
- **Switch Port** (e.g., FastEthernet0/1, FastEthernet0/2, etc.)

The hosts are organized by departments or roles, such as lab devices (lab1 to lab4), admission devices (admission_1 to admission_3), HR department devices (hr_1 to hr_3), and the administrative device **PC1**, which is being used to access this interface.

This interface provides clear visibility into host connectivity, allowing administrators to monitor the location and connection status of each endpoint on the network.

```
import json
import requests
import urllib3

# Disable SSL warnings
urllib3.disable_warnings(urllib3.exceptions.InsecureRequestWarning)

# Authentication function to get a service ticket
def get_service_ticket(username="admin", password="admin"):
    auth_url = "http://localhost:58000/api/v1/ticket"
    body = {
```

```
        "username": username,
        "password": password
    }
    response = requests.post(auth_url, json=body, verify=False)
    if response.status_code in [200, 201]:  # Accept both 200 OK and 201 Created
        return response.json()["response"]["serviceTicket"]
    else:
        print(f"Authentication failed. Status code: {response.status_code}")
        print(f"Response: {response.text}")
        return None

# Get a service ticket
service_ticket = get_service_ticket()
if not service_ticket:
    print("Failed to obtain service ticket. Exiting.")
    exit(1)

api_url = "http://localhost:58000/api/v1/host"
headers = {
    "X-Auth-Token": service_ticket
}

# Make the API request
resp = requests.get(api_url, headers=headers, verify=False)
print("Request status: ", resp.status_code)

# Check if request was successful
if resp.status_code != 200:
    print(f"API request failed. Status code: {resp.status_code}")
    print(f"Response: {resp.text}")
    exit(1)

# Parse the JSON response
try:
    response_json = resp.json()
    print(f"Response structure: {json.dumps(response_json, indent=2)[:200]}...")

    hosts = response_json.get("response", [])

    # Check if hosts is a list before iterating
    if isinstance(hosts, list):
        for host in hosts:
            if isinstance(host, dict):
                print(
                    host.get("hostName", "N/A"), "\t",
                    host.get("hostIp", "N/A"), "\t",
                    host.get("hostMac", "N/A"), "\t",
                    host.get("connectedInterfaceName", "N/A")
```

```
            )
        else:
            print(f"Unexpected host data format: {host}")
    else:
        print(f"Unexpected response format. 'response' is not a list: {hosts}")
except Exception as e:
    print(f"Error processing response: {e}")
    print(f"Raw response: {resp.text[:200]}...")
```

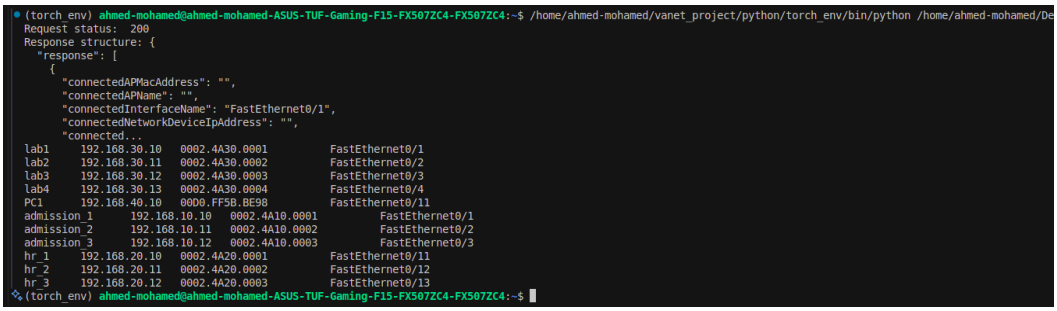
Script Overview:

We used Python to interact with the Cisco Network Controller REST API to:

- Authenticate and retrieve a **service ticket**
- Send a **GET request** to the `/api/v1/host` endpoint
- Display key details about each host (hostname, IP, MAC, and interface name)

Key Features of the Script:

- Disables SSL warnings for smoother local testing
- Handles authentication with a POST request to get the service ticket
- Automatically fetches all hosts using the ticket and displays:
  - **Host Name**
  - **IP Address**
  - **MAC Address**
  - **Interface Name**



The script successfully queried the network and received a 200 OK response, indicating a successful API call. The response includes a detailed list of connected devices with their respective hostnames, IP addresses, MAC addresses, and the switch port interfaces they are connected to. This output provides a clear mapping of network devices, useful for network inventory, monitoring, and troubleshooting purposes.

V. Conclusion

This ZeroTrustLAN project effectively implements and validates core Zero Trust Architecture principles within a Cisco Packet Tracer environment. Through meticulous configuration of VLAN segmentation, port security with explicitly defined MAC addresses, restrictive Zero-Trust ACLs, and role-based secure management (AAA/SSH), the network demonstrates robust defense against unauthorized access and limits potential threat propagation. The setup successfully enforces segmentation, the principle of least privilege, and identity verification at the network edge, providing a valuable educational model for understanding and applying ZTA concepts.

Future Enhancements & Packet Tracer File:
Save the final Cisco Packet Tracer simulation file as `ZeroTrustLAN_Project_v4_Final.pkt`.
Potential future explorations could include:

- Implementing 802.1X for dynamic, centralized authentication (would require adding a RADIUS server).
- Exploring DHCP snooping and Dynamic ARP Inspection for enhanced Layer 2 security.
- Integrating a simple firewall appliance for more advanced inter-VLAN policy enforcement.