LabCore Network Design Project

# Project Title

LabCore Network Design

# Objective / Problem Statement

The goal of this project was to design and configure a secure, segmented network for a small business using both Layer 2 and Layer 3 networking features. Within a simulated lab environment, I created VLANs to separate traffic, implemented inter-VLAN routing, configured DHCP to assign IP addresses dynamically, applied QoS to prioritize voice traffic, and added wireless access through a Wireless Access Point. I verified that all hosts had proper connectivity, phones could place internal and external calls, and devices could reach the internet.

# Project Scope

**Included:**

* Configuration of two switches
* Configuration of one router with Inter-VLAN routing and DHCP
* VLAN creation and assignment of access ports for data, voice, and management traffic
* QoS setup to prioritize VoIP traffic.
* Wireless Access Point configuration and client connectivity
* Setup and verification of user devices and IP phones

**Not Included:**

* Configuration of the internet-facing router (ISP connection was preconfigured in the lab)
* External firewall implementation or advanced security hardening

# Tools & Technologies Used

* Cisco Packet Tracer
* Cisco Catalyst Switches and Router
* IP Phones and Wireless Access Point
* Basic PC endpoints and Tablet PC

# Step-by-Step Implementation

1. Configure Core Switch, LabCore-SW1
   1. Basic Configuration
      1. #clock set hh:mm:ss DD MONTH YYYY
      2. (config)#hostname LabCore-SW1
      3. (config)#enable secret LabCorepass
      4. (config)#interface vlan 110
      5. (config-if)#ip address 192.168.110.251 255.255.255.0
      6. (config)#ip default-gateway 192.168.110.1
      7. (config)#username netadmin password LabC0re!Net
      8. (config)#service password-encryption
      9. (config)#line console 0
      10. (config-line)#login local
      11. (config)#ip access-list standard VTY
      12. (config-std-nacl)#permit 192.168.110.0 0.0.0.255
      13. (config)#line vty 0 15
      14. (config-line)#login local
      15. (config-line)#access-class VTY in
   2. Configure SSH and Disable Telnet
      1. (config)#ip domain-name labcore.local
      2. (config)#crypto key generate rsa
         1. *Modulus: 2048*
      3. (config)#ip ssh version 2
      4. (config)#line vty 0 15
      5. (config-line)#transport input ssh
      6. (config-line)#transport output ssh
   3. Configure Spanning-Tree Root Bridge
      1. (config)#spanning-tree vlan 1-1024 priority 0
   4. Create VLANs
      1. (config)#vlan 110
      2. (config-vlan)#name MGMT
      3. (config)#vlan 210
      4. (config-vlan)#name DATA
      5. (config)#vlan 160
      6. (config-vlan)#name VOICE
   5. Configure Access Ports
      1. (config)#interface fa0/1
      2. (config-if)#description NOC-PC
      3. (config-if)#switchport mode access
      4. (config-if)#switchport access vlan 110
      5. (config-if)#spanning-tree portfast
      6. *\*Connected the NOC-PC to LabCore-SW-1 with a copper straight-through cable on port fa0/1 at this point for testing*
         1. *After connecting the NOC-PC to Fa0/1 on SW1, I gave it a static IP address in the MGMT VLAN range. Since it's a management device, I didn’t use DHCP for it.*
      7. (config)#interface range fa0/2-23
      8. (config-if-range)#description Access Port
      9. (config-if-range)#switchport mode access
      10. (config-if-range)#switchport access vlan 210
      11. (config-if-range)#switchport voice vlan 160
      12. (config-if-range)#spanning-tree portfast
      13. (config-if-range)#mls qos trust cos
      14. (config-if-range)#mls qos trust device cisco-phone
   6. Turn on mls qos
      1. (config)#mls qos
   7. Configure Trunk to Router
      1. (config)#interface fa0/24
      2. (config-if)#description Trunk to Router
      3. (config-if)#switchport mode trunk
   8. Configure Trunks to LabCore-SW2
      1. (config)#interface range g0/1 – 2
      2. (config-if-range)#description Trunks to LabCore-SW2
      3. (config-if-range)#switchport mode trunk
2. Configure Second Switch, LabCore-SW2
   1. Basic Configuration
      1. #clock set hh:mm:ss DD MONTH YYYY
      2. (config)#hostname LabCore-SW2
      3. (config)#enable secret LabCorepass
      4. (config)#interface vlan 110
      5. (config-if)#ip address 192.168.110.252 255.255.255.0
      6. (config)#ip default-gateway 192.168.110.1
      7. (config)#username netadmin password LabC0re!Net
      8. (config)#service password-encryption
      9. (config)#line console 0
      10. (config-line)#login local
      11. (config)#ip access-list standard VTY
      12. (config-std-nacl)#permit 192.168.110.0 0.0.0.255
      13. (config)#line vty 0 15
      14. (config-line)#login local
      15. (config-line)#access-class VTY in
   2. Configure SSH and Disable Telnet
      1. (config)#ip domain-name labcore.local
      2. (config)#crypto key generate rsa
         1. Modulus: 2048
      3. (config)#ip ssh version 2
      4. (config)#line vty 0 15
      5. (config-line)#transport input ssh
      6. (config-line)#transport output ssh
   3. Create VLANs
      1. (config)#vlan 110
      2. (config-vlan)#name MGMT
      3. (config)#vlan 210
      4. (config-vlan)#name DATA
      5. (config)#vlan 160
      6. (config-vlan)#name VOICE
   4. Configure Access Ports
      1. (config)#interface range fa0/1 - 24
      2. (config-if-range)#description Access Port
      3. (config-if-range)#switchport mode access
      4. (config-if-range)#switchport access vlan 210
      5. (config-if-range)#switchport voice vlan 160
      6. (config-if-range)#spanning-tree portfast
      7. (config-if-range)#mls qos trust cos
      8. (config-if-range)#mls qos trust device cisco-phone
   5. Turn on mls qos
      1. (config)#mls qos
   6. Configure Trunks to LabCore-SW1
      1. (config)#interface range g0/1 - 2
      2. (config-if-range)#description Trunks to LabCore-SW1
      3. (config-if-range)#switchport mode trunk
   7. Save the configurations on both switches if not done yet (LabCore-SW1 & SW2)
      1. #copy running-config startup-config
   8. Using a crossover cable connect LabCore-SW1 G0/1 and G0/2 to LabCore-SW2 G0/1 and G0/2
3. Configure LabCore-RTR with Basic Configurations, Inter-VLAN routing, DHCP, & Internet Services
   1. Basic Configurations
      1. (config)#hostname LabCore-RTR
      2. (config)#enable secret LabCorepass
      3. (config)#username netadmin password LabC0re!Net
      4. (config)#service password-encryption
      5. (config)#line console 0
      6. (config-line)#login local
   2. Configure Interface G0/0 for Inter-VLAN routing for Mgmt, Data, & Voice VLANs
      1. (config)#interface g0/0
      2. (config-if)#no shutdown
      3. (config-if)#description Trunk to LabCore-SW1
      4. (config)#interface g0/0.110
      5. (config-subif)#description MGMT
      6. (config-subif)#encapsulation dot1q 110
      7. (config-subif)#ip address 192.168.110.1 255.255.255.0
      8. (config-subif)#ip nat inside
      9. (config)#interface g0/0.210
      10. (config-subif)#description DATA
      11. (config-subif)#encapsulation dot1q 210
      12. (config-subif)#ip address 192.168.210.1 255.255.255.0
      13. (config-subif)#ip nat inside
      14. (config)#interface g0/0.160
      15. (config-subif)#description VOICE
      16. (config-subif)#encapsulation dot1q 160
      17. (config-subif)#ip address 192.168.160.1 255.255.255.0
      18. (config-subif)#ip nat inside
   3. Using a straight-through cable connect LabCore-SW1 Fa0/24 to LabCore-RTR G0/0
   4. Configure DHCP Services for the VOICE and Data VLAN
      1. (config)#ip dhcp pool VOICE
      2. (dhcp-config)#network 192.168.160.0 255.255.255.0
      3. (dhcp-config)#default-router 192.168.160.1
      4. (dhcp-config)#dns-server 8.8.8.8
      5. (config)#ip dhcp excluded-address 192.168.160.1 192.168.160.50
      6. (config)#ip dhcp pool DATA
      7. (dhcp-config)#network 192.168.210.0 255.255.255.0
      8. (dhcp-config)#default-router 192.168.210.1
      9. (dhcp-config)#dns-server 8.8.8.8
      10. (config)#ip dhcp excluded-address 192.168.210.1 192.168.210.50
   5. Add a Default Route for Internet Access
      1. (config)#ip route 0.0.0.0 0.0.0.0 205.10.49.33
4. Connect & Configure VoIP Phones
   1. Using a straight-through cable connect Phone A to LabCore-SW1
   2. Using a straight-through cable connect Phone B to LabCore-SW2
   3. show running-config (on LabCore-RTR, to verify ephones 1 and 2 have automatically added)
5. Add Directory Numbers to the ephones on LabCore-RTR
   1. (config)#ephone-dn 1
   2. (config-ephone-dn)#number 1101
   3. (config)#ephone-dn 2
   4. (config-ephone-dn)#number 1102
   5. (config)#ephone 1
   6. (config-ephone)#mac-address 0005.5e89.d657
   7. (config-ephone)#button 1:1
   8. (config)#ephone 2
   9. (config-ephone)#mac-address 0009.7ce6.e321
   10. (config-ephone)#button 1:2
   11. \*Phones A and B should now properly register and be able to make internal and external calls
6. Save your configuration
   1. #copy running-config startup-config
7. Connect & Configure Wireless Access
   1. Using a straight-through cable connect the LabCore-WAP to port Fa0/23 on LabCore-SW1
   2. Add a description to the switchport on LabCore-SW1 to note the WAP is connected there and provide access to vlan 210
8. Configure the Wireless SSID & WPA2 PSK on LabCore-WAP
   1. LabCore-WAP->Config->Port 1
   2. SSID: Core-Lab
   3. Authentication: WPA2 PSK: corelab1
9. Configure the Tablet PC for wireless access
   1. Tablet-PC-> Config -> Wireless0
   2. SSID: Core-Lab
   3. WPA2 PSK: corelab1
      1. \*The Wireless Tablet PC should now be connected to the network, pull an address via DHCP, and have network and internet connectivity.
   4. From the desktop of Tablet-PC access the command prompt utility and perform the following
      1. C:\>ipconfig /all (to verify IP address was obtained from the DHCP server)
         1. If there is no valid IP address, run ipconfig /renew to request a new DHCP lease
      2. C:\>ping 192.168.210.1 (to verify IP connectivity to Data gateway IP on LabCore-RTR)
      3. C:\>ping 8.8.8.8 (to verify IP connectivity to the internet)
      4. C:\>ping other PCs, including the NOC-PC, to confirm full network connectivity
10. Connect User A & User B to the Network
    1. Using a straight-through cable connect User A to LabCore-SW1 on Fa0/2
       1. Verify User A is configured as a DHCP client
       2. From the desktop of User A access the command prompt utility and perform the following
          1. C:\>ipconfig /all (to verify IP address was obtained from the DHCP server)
             1. If there is no valid IP address, run ipconfig /renew to request a new DHCP lease
          2. C:\>ping 192.168.210.1 (to verify IP connectivity to Data gateway IP on LabCore-RTR)
          3. C:\>ping 8.8.8.8 (to verify IP connectivity to the internet)
          4. C:\>ping other PCs, including the Tablet and NOC-PC, to confirm full network connectivity
    2. Using a straight-through cable connect User B to LabCore-SW2
       1. Verify the User B is configured as a DHCP client
       2. From the desktop of User B access the command prompt utility and perform the following
          1. C:\>ipconfig /all (to verify IP address was obtained from the DHCP server)
             1. If there is no valid IP address, run ipconfig /renew to request a new DHCP lease
          2. C:\>ping 192.168.210.1 (to verify IP connectivity to Data gateway IP on LabCore-RTR)
          3. C:\>ping 8.8.8.8 (to verify IP connectivity to the internet)
          4. C:\>ping other PCs, including the Tablet and NOC-PC, to confirm full network connectivity
    3. From the NOC-PC desktop, open the command prompt and run:
       1. C:\>ipconfig /all (to verify the static IP address is correctly assigned)
       2. C:\>ping 192.168.110.251 (to verify connectivity to the management VLAN interface on LabCore-SW1)
       3. C:\>ping 192.168.210.1 (to verify connectivity to the data VLAN gateway on LabCore-RTR)
       4. C:\>ping 8.8.8.8 — (to verify connectivity to the internet)
       5. C:\>ping other PCs, including the Tablet, to confirm full network connectivity
11. Lab Show Commands
    1. Switch:
       1. #show vlan
       2. #show mac address-table
       3. #show interfaces trunk
       4. #show spanning-tree
       5. #show mls qos interface <interface>
    2. Router:
       1. #show ip route
       2. #show ip dhcp binding
       3. #show ip nat translations
       4. #show ephone
       5. #show policy-map interface <interface>
    3. Both:
       1. #show startup-config
       2. #show ip interface brief
       3. #show interface <interface>
       4. #show clock
       5. #show ssh
       6. #show access-list
       7. #show running-config
       8. #show running-config | begin <word>
       9. #show running-config | include <word>

# Challenges and Solutions

* **Ephones not registering:**
  + Issue: Phones were receiving the error: Received dn 1 exceeds max-dn 0 and ephone exceeds max-ephone 0 when verifying ephones on the router.
  + Solution: Updated max-dn and max-ephones to 10 under telephony-service.
    - Set the correct ip source-address to match the router's interface IP (e.g., 192.168.160.1).
  + Result: Phones successfully registered and were assigned their directory numbers.
* PSTN Ephone not registering
  + Issue: During testing, ephones couldn’t make outbound calls because the PSTN phone wasn’t registering properly. This was caused by exceeding the configured maximum number of ephones and ephone-dns (max ephones and max dn). Additionally, the dial-peer configurations had been removed or lost. The issue was resolved by re-adding the ephones and ephone-dns within the current configured limits, reapplying the dial-peer configurations, and re-entering the IP source address on the PSTN router.
  + Result: Ephones can make outbound calls with PSTN phone
* VLAN 110 incorrect on SW1
  + Issue: Put VLAN 100, updated to VLAN 110 to correct the network to ensure it works properly.
* Typo in VLAN Command
  + Issue: While creating VLANs on the second switch, I accidentally typed vlal 160 instead of vlan 160, which returned a syntax error. I reviewed the command, corrected the typo, and successfully created VLAN 160 named VOICE.

# Project Outcomes / Results

The project met all the initial objectives. VLANs successfully segmented traffic by type, and inter-VLAN routing allowed communication across VLANs. DHCP functioned correctly, dynamically assigning addresses within the appropriate subnet. VoIP phones were able to register and place calls, and the wireless client was able to connect, receive a DHCP address, and browse the internet. Overall, the simulated environment mirrored real-world setups effectively.

# Screenshots / Diagrams

1. Topology Diagrams
   1. Starting Topology

A computer screen shot of a computer network

AI-generated content may be incorrect.

* 1. Completed/Final Topology

A computer screen shot of a computer network

AI-generated content may be incorrect.

* 1. All Topology Diagrams
     1. [View Topology Folder](https://github.com/baybiegirl20/IT-Projects/tree/Networking-Projects/LabCore%20Network%20Design/LabCore%20Network%20Design%20Screenshots/Topology)

1. Switch Configurations
   1. SW1 Startup Configuration

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A white background with black dots

AI-generated content may be incorrect.

* + - 1. [View Switch 1 Folder](https://github.com/baybiegirl20/IT-Projects/tree/Networking-Projects/LabCore%20Network%20Design/LabCore%20Network%20Design%20Screenshots/Switch%201)
  1. SW2
     1. Not included in document (configuration similar to SW1)
        1. [View Switch 2 Folder](https://github.com/baybiegirl20/IT-Projects/tree/Networking-Projects/LabCore%20Network%20Design/LabCore%20Network%20Design%20Screenshots/Switch%202)

1. Router Configuration
   1. Router Startup Configuration

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

* + - 1. [View Router Folder](https://github.com/baybiegirl20/IT-Projects/tree/Networking-Projects/LabCore%20Network%20Design/LabCore%20Network%20Design%20Screenshots/Router)

1. Troubleshooting
   1. Typo in VLAN

A close-up of a person

AI-generated content may be incorrect.

* 1. Wrong VLAN number used

A screenshot of a computer program

AI-generated content may be incorrect.

* 1. Additional troubleshooting steps and screenshots can be found here:
     1. [View Troubleshooting Folder](https://github.com/baybiegirl20/IT-Projects/tree/Networking-Projects/LabCore%20Network%20Design/LabCore%20Network%20Design%20Screenshots/Troubleshooting)

1. Testing & Verification
   1. Connectivity Testing
      1. User A

A computer screen shot of a black screen

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

* + 1. Phones

A screenshot of a phone

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

* + 1. [View Test & Verify Folder](https://github.com/baybiegirl20/IT-Projects/tree/Networking-Projects/LabCore%20Network%20Design/LabCore%20Network%20Design%20Screenshots/Test%20%26%20Verify)

1. Show Commands (All Devices)
   1. [View Show Commands Folder](https://github.com/baybiegirl20/IT-Projects/tree/Networking-Projects/LabCore%20Network%20Design/LabCore%20Network%20Design%20Screenshots/Show%20Commands)

# Conclusion / Lessons Learned

This project gave me firsthand experience with core networking tasks in a simulated environment that closely resembles real-world scenarios. It strengthened my understanding of how to build secure and scalable LANs, and helped me troubleshoot key services like DHCP, QoS, and inter-VLAN routing.

# GitHub or Project Link

[Github Link](https://github.com/baybiegirl20/IT-Projects/tree/Networking-Projects/LabCore%20Network%20Design/)