# Network infrastructure

## INDEX

# Data Centre Strategy.

Data centre hardware includes a broad range of devices. Hardware components that make up the IT infrastructure of a data centre include servers and computing equipment, networking equipment and storage equipment, plus additional infrastructure to support it like power and cooling systems. Get insights into developing a data centre hardware strategy.

One of the most popular network lifecycle models is Cisco's PPDIOO (Prepare, Plan, Design, Implement, Operate and Optimize) model:

- Prepare. This is where you define high-level requirements and strategy. For example, your deliverables from this phase may include requirements documentation and current state surveys.
- Plan. This stage deals with specific network requirements based on information gathered in the planning stages.
- Design. During the design stage, the information gathered from the previous two stages is used to create a detailed network design.
- Implement. This is where the work gets done to configure and deploy the network infrastructure. There is often testing to validate the design in this phase.

- Operate. This is the portion of the lifecycle where the network is in production use. During this stage, monitoring is an important part of validating that the network is working as designed and being able to quickly address issues when it isn't.
- Optimize. At some point in most networks' lifecycle, tweaks and optimizations are needed. This is the stage where those changes are identified. For major changes, the cycle begins again to plan and implement them.

**Consider the needs of the end-user**

Today's data centres are designed to be more resilient and reliable. This means catering to the needs of different end users. Different data, desktops and applications should be accessed by end users from anywhere and using any device. By investing in a modern data centre, the organization should ensure that it meets the wide range of workload demands.

**Use hybrid cloud environments**

Hybrid environments that are managed by the data centre team can help in controlling resources and improving security and performance. Other than using a public cloud to keep critical applications running efficiently, private clouds can be set up and managed internally. This facilitates resource sharing across multiple teams.

**Service continuity**

It is important for data centres to provide near 100% availability rather than just taking time to come up with the best disaster recovery strategies. Distributing applications across multiple sites helps to improve performance and increase uptime. It's the same model that Facebook, Amazon and Google among others have used and achieved great success.

# INSIGHTS ON BUILDING of A DATA CENTER

The design must allow for people to be successful within the environment. You cannot create an unreliable infrastructure and place the burden of expectations of reliability on your IT staff to manage it.

- The design should be as safe and straightforward as possible.
- The design should be generally fault-tolerant
- The design should be able to scale reasonably

# STEPS, CONSIDERATIONS & SPECIFICATIONS FOR BUILDING A DATA CENTER

**Backup power and generation**

Range anxiety is a real issue in smaller/remote data centres and coordinating fuel delivery logistics for smaller facilities can be a burden. For this reason, we select diverse fuel sources for our diverse generator systems. One generator should be diesel, while the other should be natural gas.

Suppose an entire building is backed up by a backup generation system. For example, suppose the primary building generator system is natural gas. In that case, the backup will be diesel for the other If the primary building generator system is diesel the backup will be natural gas for the other.

**Switching and distribution**

All inbound feeds from the exterior to the datacentre (exterior utilities, mechanical load that is outside of the room, feeds from the main building) should have a transient voltage surge suppression system in line with the main feed. The mechanical panel may also have a TVSS system in line.

**UPS systems (Uninterruptible Power Supplies)**

Double Online Conversion systems are the preferred system to be used. By choice, or for small data centres, in-rack UPS systems are acceptable. Each rack should have two double online conversion systems fed from each of the upstream power sources. They should not parallel in any way, shape, or form. Dedicated stand-alone units should be in separate electrical rooms.

As with all other distribution elements, the units should experience no greater than 80% loaded under the worst-case failure scenario while fully loaded. There is a minimum of two separate UPS systems that are each fed by one of the two different utility feeds.

Large data centres take advantage of the economies of scale with very robust free-standing units. Most people's favourite is freestanding unit is the Toshiba G9000. As you can see, these START to become relevant at around 100 kVA and then grow from there. I have used the 750 KVA G9000’s for many years and love them.

**Cooling units' installation**

The best classic system for a reliable facility is DX systems with individual condenser feeds along separate paths to an exterior location for heat rejection. The reliability of the chiller, whether dedicated to the data centre or building, should not contribute to the system’s overall reliability under this design. While chillers do factually provide a cooling source, we consider it a utility for purposes of a reference architecture for reasons I discuss below.

Building maintenance teams are often not sensitive to the unique requirements of the data centre environment. Many times, I have seen stories from end-users about the chillers shut off for maintenance with no prior notice.

Standard guidance for designing nuclear plants and submarines is to minimize the number of valves used. DX units can have as few as zero valves, while chillers require a complex symphony of valves to meet minimally viable concurrent maintainability. There must be a good reason that this is a topic of discussion in nuclear plant design?

All mechanical systems should be sufficiently elevated, and gravity fed to either interior or exterior drainage for the condensate pan. Each system should have an external leak detection sensor ringed around it in the event of water leaks. If a condensate pan must feed via a pump, the system must have a dual float off switch, be fed via redundant condensate pumps, and have an elevated floor dam built around it with leak sensors.

**Fire suppression**

Dual interlock or gas-based fire suppression systems are permissible. Careful about the management of the environment with gas and VESDA, these systems are notorious for false alarms and life safety issues.

**Building & environmental monitoring system**

UPS, and all mechanical systems (all active capacity components) shall monitor via a building management system. Environmental sensors throughout the facility provide insight into the ambient temperature. BMS shall have a daily heartbeat that will send an email to responsible parties.

**Security systems**

Security is considered an ancillary system from the perspective of reliability site engineering. Still, each system at the point of use shall be sufficiently secure to meet the organization’s overall requirements.

**Lightning protection**

UL Listed Lightning protection dictates a full building set of air terminals, to include any exterior equipment. Lightning protection is good practice in any building, including one in which a data centre resides. In addition to that, any inbound feeds from exterior utilities should have transient voltage surge suppression to include utility feeds from the rest of the building to the controlled data centre environment.

**Leak protection**

The datacentre is preferably underneath sufficient roofing to prevent any types of leaks. Rooftop mounted equipment and penetrations account for 70% of leaks, especially in storms.

**Containment and air handling**

Hot aisle containment is the preferred methodology for the comfort of the room and pre-cooling volumetric space for redundancy ride through purposes during transient failures. In the event of large clearance space, the containment may be partial with drop ceiling and ported return grates in the ceiling. The HVAC / CRAC Units may be ducted to the ceiling’s return and draw the hot air back that way. In the event this is not possible, a thermally secure containment system must be built with common ducting back to the HVAC systems. A common plenum hung from the ceiling and tied into a full or partial containment will work in areas with less clearance.

Hot aisle containment is also preferable because it prevents batteries and ancillary systems from being unnecessarily exposed to heat if they are in the same space, which may be the case for small data centres. The ideal operating temperature for batteries is mid 70's, any 10-degree Celsius variation from this will cause a 50% reduction in battery life.

Alternatively, in-row cooling systems that utilize DX are acceptable and even desirable in tight areas that do not have clearance. Containment is essential here. Be careful to coordinate containment with fire protection. Many fire marshals require either unique sprinklers in the contained space or some ability through fusible links to drop or shrink the containment in the event of a fire.

# DESIGN

in this design we have the following:

- 2 ISP - ROUTER 2811
- 2 core routers.
- 2 multilayer switches
- 10 Department so 10 switch 2960
- For connections
  - Routers serial connection
  - Router to multilayer switch normal cable,
  - Multilayer switches to access layer cross over cables
- for the core router and ISP router, there is no serial port
  - so, click the router > physical > off the router and drag HWIC-2T to an empty slot and turn on the router.
- SWITCH -> TO ADD POWER > DRAG AC-POWER-SUPPLY
- ROUTER -> TO ON > CONFIG > INTERFACE (GIGABITETHERNET0/ .. ETC) > TICK ON
- All the devices are connected two times cause of the redundancies
- (Network redundancy is a communications pathway that has additional links to connect all nodes in case one link goes down.)

# CONFIGURATION STEPS

## 1. Basic setting to all devices, ssh on the router and switches

**CONFI FOR THE FLOOR SWITCH**

en
conf t
hostname SERVERROOM-SW
line console 0

password cisco
login
exit

enable password cisco
no ip domain-lookup
banner motd #NO unauthorised access!!!#
service  password-encryption

do wr

**CONFI FOR THE MULTILAYER SWITCH**

en

conf t

hostname CORE-R2

line console 0

password cisco

login

exit

enable password cisco

no ip domain-lookup

banner motd #NO unauthorised access!!!#

service  password-encryption

do wr

**CONFIGURATION THE SLL**

- Always make sure that the switch has the host name
- Always confi the ip domain name

ip domain name cisco.net

username admin password cisco

crypto key generate rsa

1024

line vty 0 15

login local

transport input ssh

exit

do wr

en

config t

ip ssh version 2

do wr

## 2. VLANS ASSIGNMENT, ALL ACESS, TRUCK PORT ON 12,13 SWITCHES

- So what is Truck Port – it is the connection on a switch that is used to connect a guest virtual machine that is VLAN aware
- We used two vlan port, first to acesse the used port and second to access the unused port

**For the floor switches**

Int range fa0/1-2

Switchport mode trunk

Exit

Vlan 60

Name SERVERROOM

Vlan 99

Name BlackHole

exit

Int range fa0/3-24

Switchport mode access

Switchport access vlan 60

Exit

Int range gig0/1-2

Switchport mode access

Switchport access vlan 99

Exit

Do wr

## 3. SUBNETTING AND IP ADDRESS

- Making the port trunk in the multilayer switch

Int range gig1/0/3-8

Switchport mode trunk

Vlan 10

Name SALES

Vlan 20

Name HR

Vlan 30

Name FINANCE

Vlan 40

Name ADMIN

Vlan 50

Name ICT

Vlan 60

Name SERVERROOM

EXIT

Do wr

**BETWEEN THE ROUTER AND LAYER 3 SWITCHES MULTILAYER**

| NO | NETWORK ADDRESS | SUBNET MASK | HOST ADDRESS RANGE | BROADCAST ADDRESS |
|---|---|---|---|---|
| **R1-MLSW1** | 172.16.3.144 | 255.255.255.252 | 172.16.3.145 TO 172.16.3.146 | 172.16.3.147 |
| **R1-MLSW2** | 172.16.3.148 | 255.255.255.252 | 172.16.3.149 TO 172.16.3.150 | 172.16.3.151 |
| **R2-MLSW1** | 172.16.3.152 | 255.255.255.252 | 172.16.3.153 TO 172.16.3.154 | 172.16.3.155 |
| **R2-MLSW2** | 172.16.3.156 | 255.255.255.252 | 172.16.3.157 TO 172.16.3.158 | 172.16.3.159 |

**Between the router and the isps**

Public ip address

- 195.136.17.0 /30
- 195.136.17.4 /30
- 195.136.17.8 /30
- 195.136.17.12 /30

Now we will IMPLEMENT THE DESIGNETED IP ADDRESS

**IN MULTILAYER SWITCH**

- **MULTILAYER SWITCH to CORE R1 –R2 = 1G0/1-2**

Int range gig1/0/1 - 2
No switchport
Exit
Do wr

**No let's assighn those interface with ip address meaning they are now routable**

**In Multilayer switch1**

Int gig1/0/2
Ip address 172.16.3.149 255.255.255.252
No sh

Exit
Do wr

```
MLT-SW1(config)#Int gig1/0/1
MLT-SW1(config-if)#
MLT-SW1(config-if)#Ip address 172.16.3.145 255.255.255.252
MLT-SW1(config-if)#
MLT-SW1(config-if)#No sh
MLT-SW1(config-if)#
MLT-SW1(config-if)#Exit
MLT-SW1(config)#
MLT-SW1(config)#
MLT-SW1(config)#
MLT-SW1(config)#Do wr
Building configuration...
Compressed configuration from 7383 bytes to 3601 bytes[OK]
[OK]
```

```
MLT-SW1(config)#
MLT-SW1(config)#Int gig1/0/2
MLT-SW1(config-if)#
MLT-SW1(config-if)#Ip address 172.16.3.149 255.255.255.252
MLT-SW1(config-if)#
MLT-SW1(config-if)#No sh
MLT-SW1(config-if)#
MLT-SW1(config-if)#Exit
MLT-SW1(config)#
MLT-SW1(config)#
MLT-SW1(config)#
MLT-SW1(config)#Do wr
Building configuration...
Compressed configuration from 7383 bytes to 3601 bytes[OK]
[OK]
```

**Multilayer Switch2**

- **Gig 0/1 -> 172.16.3.152/30**
- **Gig 0/2 -> 172.16.3.156/30**

Int gig1/0/1
Ip address 172.16.3.153 255.255.255.252
No sh
Exit
Do wr

```
MLT-SW2(config)#Int gig1/0/1
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ip address 172.16.3.153 255.255.255.252
MLT-SW2(config-if)#
MLT-SW2(config-if)#No sh
MLT-SW2(config-if)#
MLT-SW2(config-if)#Exit
MLT-SW2(config)#
MLT-SW2(config)#
MLT-SW2(config)#
MLT-SW2(config)#Do wr
Building configuration...
Compressed configuration from 7383 bytes to 3601 bytes[OK]
```

```
MLT-SW2(config)#Int gig1/0/2
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ip address 172.16.3.157 255.255.255.252
MLT-SW2(config-if)#
MLT-SW2(config-if)#No sh
MLT-SW2(config-if)#
MLT-SW2(config-if)#Exit
MLT-SW2(config)#
MLT-SW2(config)#
MLT-SW2(config)#
MLT-SW2(config)#Do wr
Building configuration...
Compressed configuration from 7383 bytes to 3601 bytes[OK]
[OK]
```

**IN CORE R1**

- **GIG 0/0 -> 172.16.3.144 /30**
- **GIG 0/1 -> 172.16.3.152/30**
- **SE0/3/0 -> 195.136.17.0/30**
- **SE0/3/1-> 195.136.17.4/30**

Int gig0/0
Ip address 172.16.3.146 255.255.255.252
No sh
Exit
Do wr

Int gig0/1
Ip address 172.16.3.154 255.255.255.252 **(154 CAUSE 153 WAS TAKED BY MLT SW2)**
No sh
Exit
Do wr

```
CORE-R1(config)#Int gig0/0
CORE-R1(config-if)#
CORE-R1(config-if)#Ip address 172.16.3.146 255.255.255.252
CORE-R1(config-if)#
CORE-R1(config-if)#No sh
CORE-R1(config-if)#
CORE-R1(config-if)#Exit
CORE-R1(config)#
CORE-R1(config)#
CORE-R1(config)#
CORE-R1(config)#Do wr
Building configuration...
[OK]
CORE-R1(config)#Int gig0/1
CORE-R1(config-if)#
CORE-R1(config-if)#Ip address 172.16.3.154 255.255.255.252
CORE-R1(config-if)#
CORE-R1(config-if)#No sh
CORE-R1(config-if)#
CORE-R1(config-if)#Exit
CORE-R1(config)#
CORE-R1(config)#
CORE-R1(config)#
CORE-R1(config)#Do wr
Building configuration...
[OK]
```

INT SE0/3/0
Clock rate 64000
Ip add 195.136.17.1 255.255.255.252
No sh
Exit

INT SE0/3/1
Clock rate 64000
Ip add 195.136.17.5 255.255.255.252
No sh
Exit
Do wr

```
CORE-R1(config)#INT SE0/3/0
CORE-R1(config-if)#
CORE-R1(config-if)#Clock rate 64000
CORE-R1(config-if)#
CORE-R1(config-if)#Ip add 195.136.17.1 255.255.255.252
CORE-R1(config-if)#
CORE-R1(config-if)#No sh
CORE-R1(config-if)#
CORE-R1(config-if)#Exit
CORE-R1(config)#INT SE0/3/1
CORE-R1(config-if)#
CORE-R1(config-if)#Clock rate 64000
CORE-R1(config-if)#
CORE-R1(config-if)#Ip add 195.136.17.5 255.255.255.252
CORE-R1(config-if)#
CORE-R1(config-if)#No sh
CORE-R1(config-if)#
CORE-R1(config-if)#Exit
CORE-R1(config)#
CORE-R1(config)#Do wr
Building configuration...
[OK]
```

**IN CORE R2**

- **GIG 0/0 -> 172.16.3.148 / 30**
- **GIG 0/1 -> 172.16.3.156/30**
- **SE0/3/0 -> 195.136.17.8 /30**
- **SE0/3/1 -> 195.136.17.12/30**

Int gig0/0
Ip address 172.16.3.150 255.255.255.252
No sh
Exit
Do wr

Int gig0/1
Ip address 172.16.3.158 255.255.255.252
No sh
Exit
Do wr

```
CORE-R2(config)#Int gig0/0
CORE-R2(config-if)#
CORE-R2(config-if)#Ip address 172.16.3.150 255.255.255.252
CORE-R2(config-if)#
CORE-R2(config-if)#No sh
CORE-R2(config-if)#
CORE-R2(config-if)#Exit
CORE-R2(config)#
CORE-R2(config)#
CORE-R2(config)#
CORE-R2(config)#Do wr
Building configuration...
[OK]
CORE-R2(config)#
CORE-R2(config)#Int gig0/1
CORE-R2(config-if)#
CORE-R2(config-if)#Ip address 172.16.3.158 255.255.255.252
CORE-R2(config-if)#
CORE-R2(config-if)#No sh
CORE-R2(config-if)#
CORE-R2(config-if)#Exit
CORE-R2(config)#
CORE-R2(config)#
CORE-R2(config)#
CORE-R2(config)#Do wr
Building configuration...
```

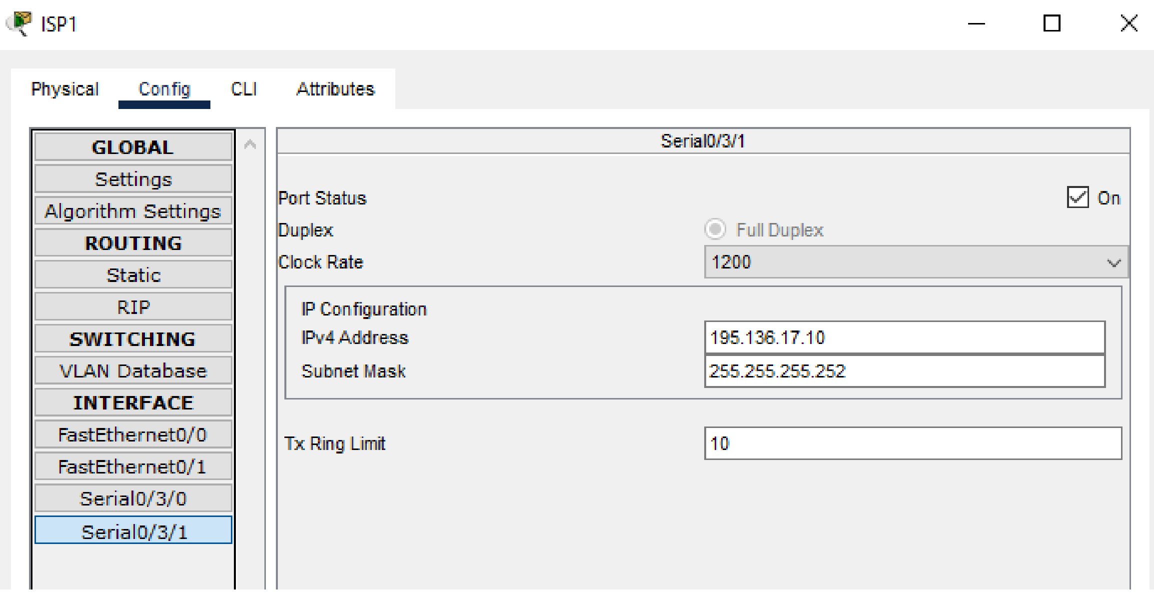
INT SE0/3/0
Clock rate 64000
Ip add 195.136.17.9 255.255.255.252
No sh
Exit

INT SE0/3/1
Clock rate 64000
Ip add 195.136.17.13 255.255.255.252
No sh
Exit
Do wr

```
CORE-R2(config)#INT SE0/3/0
CORE-R2(config-if)#
CORE-R2(config-if)#Clock rate 64000
CORE-R2(config-if)#
CORE-R2(config-if)#Ip add 195.136.17.9 255.255.255.252
CORE-R2(config-if)#
CORE-R2(config-if)#No sh
CORE-R2(config-if)#
CORE-R2(config-if)#Exit
CORE-R2(config)#
CORE-R2(config)#
CORE-R2(config)#
CORE-R2(config)#INT SE0/3/1
CORE-R2(config-if)#
CORE-R2(config-if)#Clock rate 64000
CORE-R2(config-if)#
CORE-R2(config-if)#Ip add 195.136.17.13 255.255.255.252
CORE-R2(config-if)#
CORE-R2(config-if)#No sh
CORE-R2(config-if)#
CORE-R2(config-if)#Exit
CORE-R2(config)#
CORE-R2(config)#Do wr
Building configuration...
[OK]
```

**In ISP1**

- IN CONFIG > SERIAL0/3/0 > IPv4 - 195.136.17.2, SUBNET MASK 255.255.255.252.
- IN CONFIG > SERIAL0/3/01> IPv4 - 195.136.17.10, SUBNET MASK 255.255.255.252.
- IN CLI > DO WR.



ISP1

Physical Config CLI Attributes

IOS Command Line Interface

```
Router>enable
Router#
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface Serial0/3/0
Router(config-if)#ip address 195.136.17.2 255.255.255.0
Router(config-if)#ip address 195.136.17.2 255.255.255.252
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial0/3/1
Router(config-if)#ip address 195.136.17.10 255.255.255.252
Router(config-if)#ip address 195.136.17.10 255.255.255.252
Router(config-if)#DO WR
Building configuration...
[OK]
Router(config-if)#
```

**In ISP2**

- IN CONFIG > SERIAL0/3/0 > IPv4 - 195.136.17.6, SUBNET MASK 255.255.255.252.
- IN CONFIG > SERIAL0/3/01> IPv4 - 195.136.17.14, SUBNET MASK 255.255.255.252.
- IN CLI > DO WR.

ISP2

Physical Config CLI Attributes

IOS Command Line Interface

```
Router>enable
Router#
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface Serial0/3/0
Router(config-if)#
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial0/3/0
Router(config-if)#ip address 195.136.17.6 255.255.255.0
Router(config-if)#ip address 195.136.17.6 255.255.255.252
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial0/3/1
Router(config-if)#ip address 195.136.17.14 255.255.255.252
Router(config-if)#ip address 195.136.17.14 255.255.255.252
Router(config-if)#DO WR
Building configuration...
[OK]
```

**In multilayer switch**

Ip route 0.0.0.0 0.0.0.0 gig1/0/1
Ip route 0.0.0.0 0.0.0.0 gig1/0/2 70
Do wr

```
MLT-SW1(config)#Ip route 0.0.0.0 0.0.0.0 gig1/0/1
MLT-SW1(config)#
MLT-SW1(config)#Ip route 0.0.0.0 0.0.0.0 gig1/0/2 70
MLT-SW1(config)#
MLT-SW1(config)#Do wr
Building configuration...
Compressed configuration from 7383 bytes to 3601 bytes[OK]
[OK]
```

```
MLT-SW2(config)#Ip route 0.0.0.0 0.0.0.0 gig1/0/1
MLT-SW2(config)#
MLT-SW2(config)#Ip route 0.0.0.0 0.0.0.0 gig1/0/2 70
MLT-SW2(config)#
MLT-SW2(config)#Do wr
Building configuration...
Compressed configuration from 7383 bytes to 3601 bytes[OK]
[OK]
```

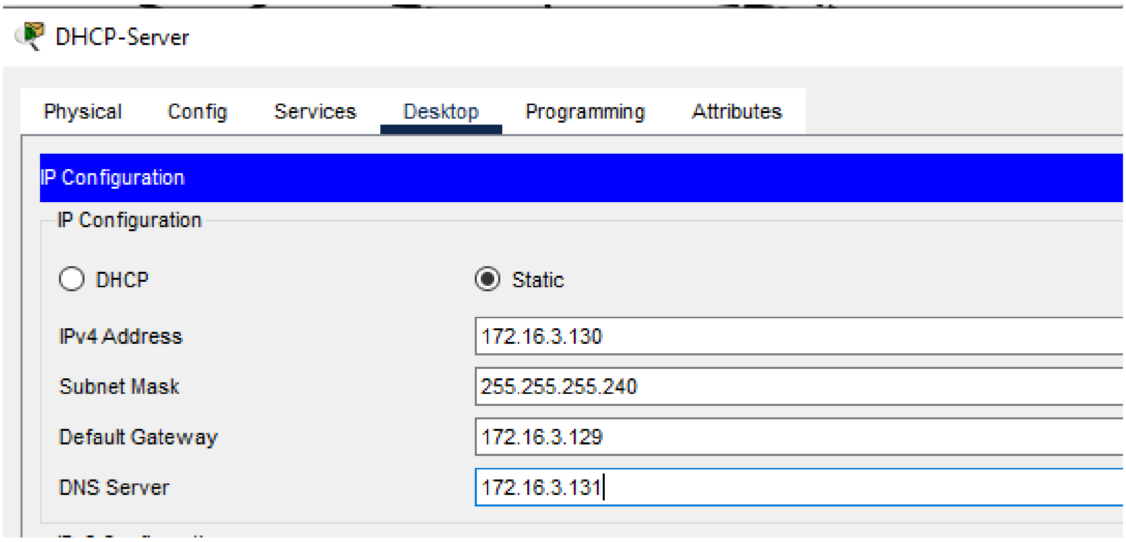
**For the core router**

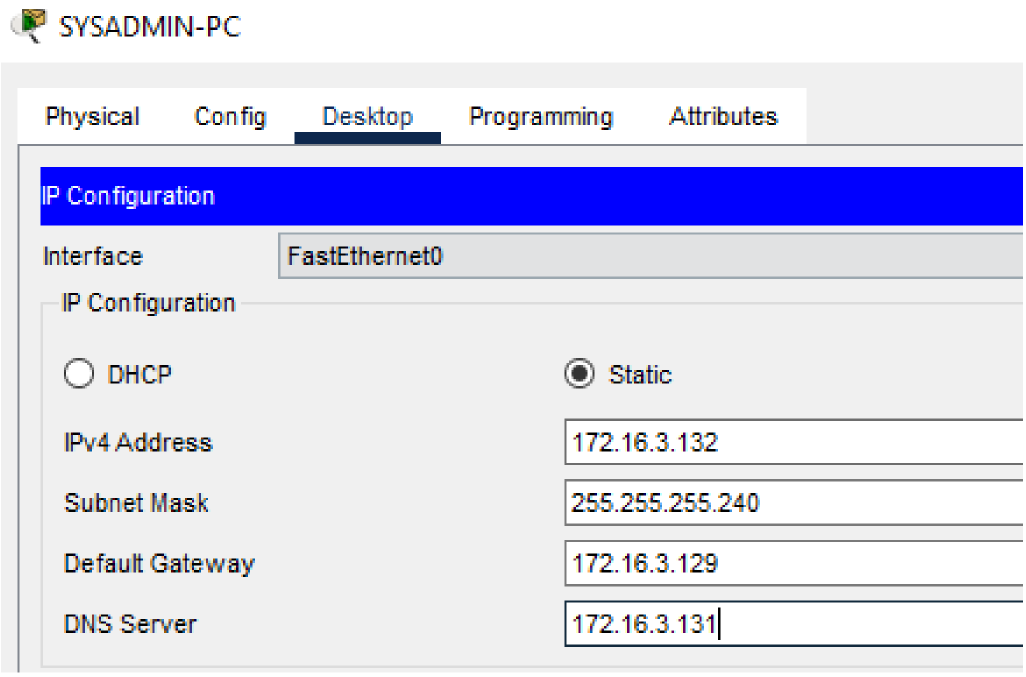
Ip route 0.0.0.0 0.0.0.0 se0/3/0
Ip route 0.0.0.0 0.0.0.0 se0/3/1 70
Do wr

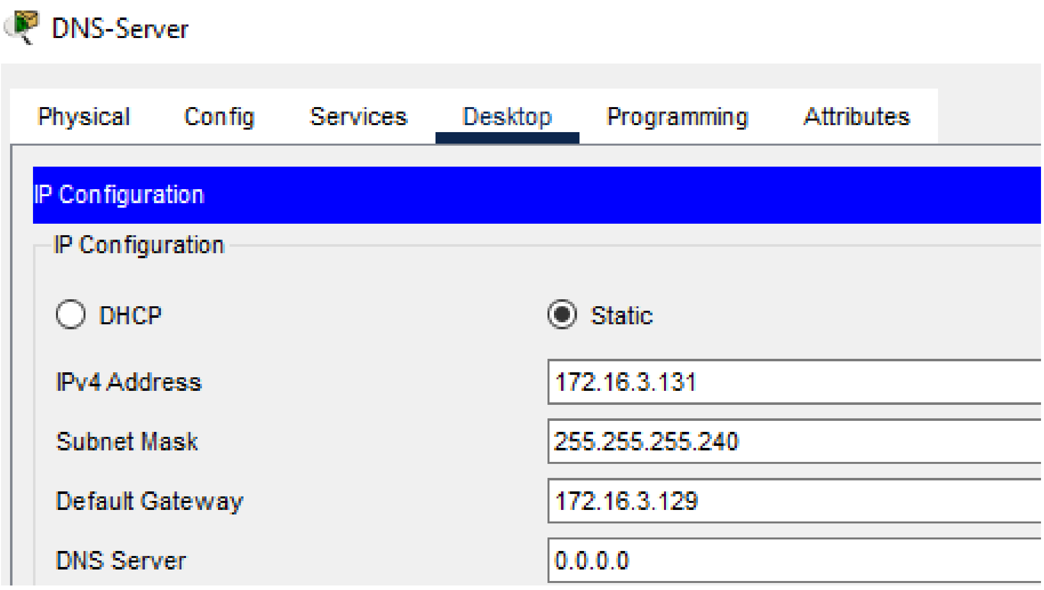
```
CORE-R1(config-if)#ex
CORE-R1(config)#Ip route 0.0.0.0 0.0.0.0 se0/3/0
%Default route without gateway, if not a point-to-point interface, may impact performance
CORE-R1(config)#
CORE-R1(config)#Ip route 0.0.0.0 0.0.0.0 se0/3/1 70
CORE-R1(config)#
CORE-R1(config)#Do wr
Building configuration...
[OK]
```

```
CORE-R2(config)#Ip route 0.0.0.0 0.0.0.0 se0/3/0
%Default route without gateway, if not a point-to-point interface, may impact performance
CORE-R2(config)#
CORE-R2(config)#Ip route 0.0.0.0 0.0.0.0 se0/3/1 70
CORE-R2(config)#
CORE-R2(config)#Do wr
Building configuration...
[OK]
```

4. STATIC IP ADDRESS TO THE SERVER ROOM DEVICES

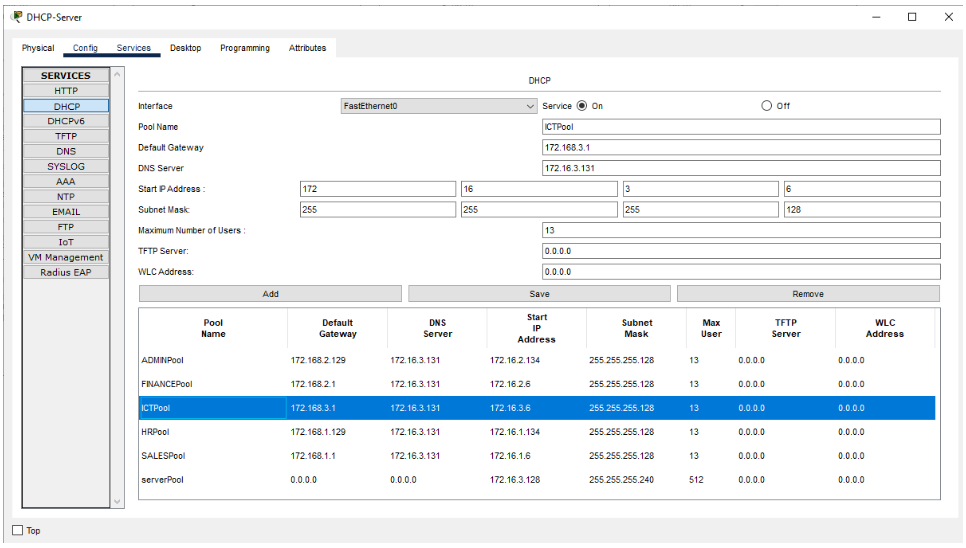

DHCP-Server
Physical
Config
Services
Desktop
Programming
Attributes
IP Configuration
IP Configuration
DHCP
Static
IPv4 Address
172.16.3.130
Subnet Mask
255.255.255.240
Default Gateway
172.16.3.129
DNS Server
172.16.3.131

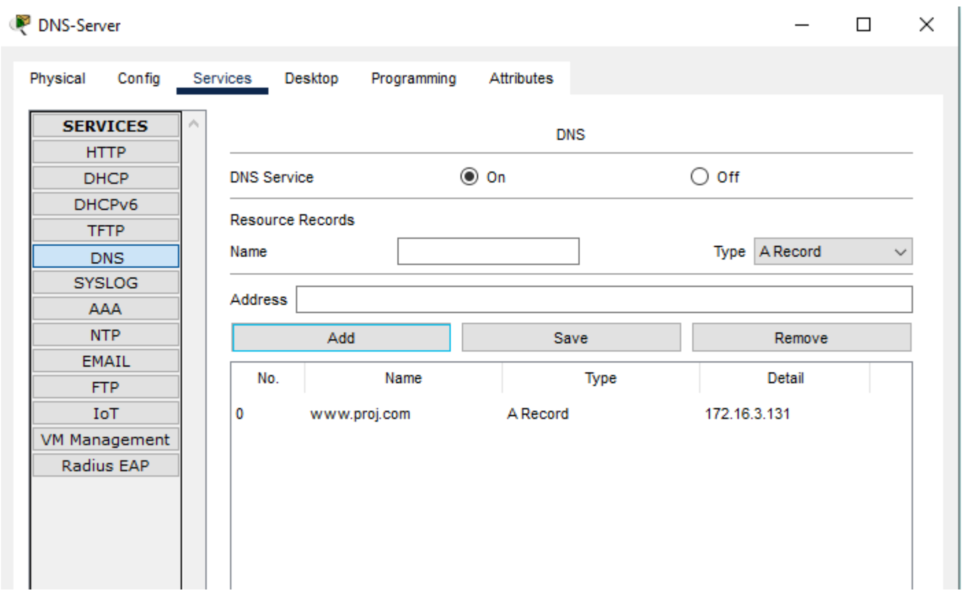

SYSADMIN-PC
Physical
Config
Desktop
Programming
Attributes
IP Configuration
Interface
FastEthernet0
IP Configuration
DHCP
Static
IPv4 Address
172.16.3.132
Subnet Mask
255.255.255.240
Default Gateway
172.16.3.129
DNS Server
172.16.3.131


DNS-Server
Physical
Config
Services
Desktop
Programming
Attributes
IP Configuration
IP Configuration
DHCP
Static
IPv4 Address
172.16.3.131
Subnet Mask
255.255.255.240
Default Gateway
172.16.3.129
DNS Server
0.0.0.0

## 5. DHCP SERVER DEVICE CONFIG

- WE NEED FROM THE OTHER ROOMS TO GET THE IP ADDRESS AUTOMATICLA SO HE HAVE TO CONFIG THE DHCP.
- ADDING OTHER NETWORKD IN THE DHCP SERVER .
- The server area didn't have a pool because it was made static.



## 6. INTER-VLAN ROOTING ON THE 13 SWITCHES PLUS IP DHCP HELPER ADDRESS

**In Multilayer switch 1 and 2**

Int vlan 10

No sh
Ip add 172.16.1.1 255.255.255.128
Ip helper-address 172.16.3.130 (the dhcp ip address)
ex

Int vlan 20

No sh
Ip add 172.16.1.129 255.255.255.128
Ip helper-address 172.16.3.130
ex

Int vlan 30

No sh
Ip add 172.16.2.1 255.255.255.128
Ip helper-address 172.16.3.130
ex

Int vlan 40

No sh
Ip add 172.16.2.129 255.255.255.128
Ip helper-address 172.16.3.130
ex

Int vlan 50

No sh
Ip add 172.16.3.1 255.255.255.128
Ip helper-address 172.16.3.130
Ex

Int vlan 60

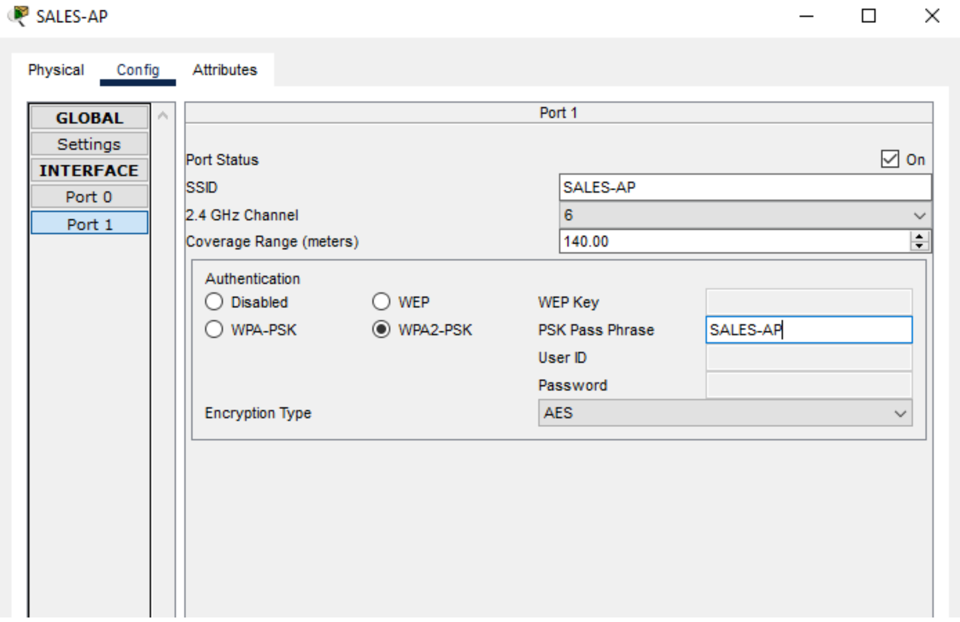
No sh
Ip add 172.16.3.129 255.255.255.240

Ip helper-address 172.16.3.130
ex
Do wr

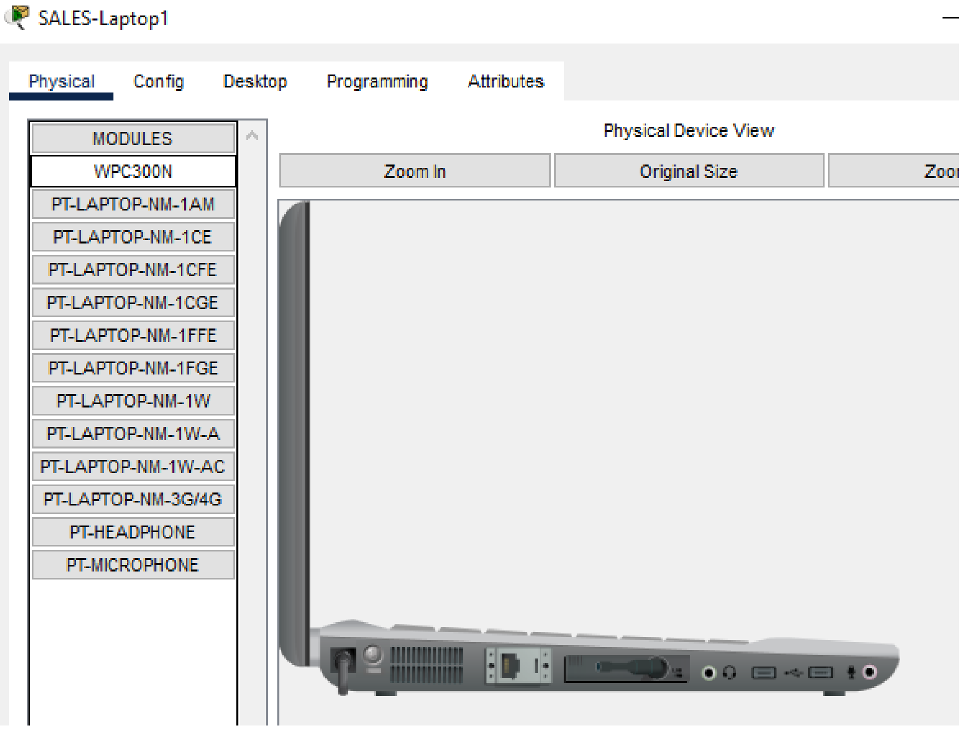
```
MLT-SW1(config)#Int vlan 20
MLT-SW1(config-if)#
MLT-SW1(config-if)#
MLT-SW1(config-if)#
MLT-SW1(config-if)#No sh
MLT-SW1(config-if)#
MLT-SW1(config-if)#Ip add 172.16.1.129 255.255.255.128
MLT-SW1(config-if)#
MLT-SW1(config-if)#Ip helper-address 172.16.3.130
MLT-SW1(config-if)#
MLT-SW1(config-if)#ex
MLT-SW1(config)#
MLT-SW1(config)#
MLT-SW1(config)#
MLT-SW1(config)#Int vlan 30
MLT-SW1(config-if)#
MLT-SW1(config-if)#
MLT-SW1(config-if)#
MLT-SW1(config-if)#No sh
MLT-SW1(config-if)#
MLT-SW1(config-if)#Ip add 172.16.2.1 255.255.255.128
MLT-SW1(config-if)#
MLT-SW1(config-if)#Ip helper-address 172.16.3.130
MLT-SW1(config-if)#
MLT-SW1(config-if)#ex
MLT-SW1(config)#
MLT-SW1(config)#
MLT-SW1(config)#
MLT-SW1(config)#Int vlan 40
MLT-SW1(config-if)#
MLT-SW1(config-if)#
MLT-SW1(config-if)#
MLT-SW1(config-if)#No sh
MLT-SW1(config-if)#
MLT-SW1(config-if)#Ip add 172.16.2.129 255.255.255.128
MLT-SW1(config-if)#
MLT-SW1(config-if)#Ip helper-address 172.16.3.130
MLT-SW1(config-if)#
MLT-SW1(config-if)#ex
MLT-SW1(config)#
MLT-SW1(config)#
MLT-SW1(config)#
MLT-SW1(config)#Int vlan 50
MLT-SW1(config-if)#
MLT-SW1(config-if)#
MLT-SW1(config-if)#
MLT-SW1(config-if)#No sh
MLT-SW1(config-if)#
MLT-SW1(config-if)#Ip add 172.16.3.1 255.255.255.128
MLT-SW1(config-if)#
MLT-SW1(config-if)#Ip helper-address 172.16.3.130
MLT-SW1(config-if)#
MLT-SW1(config-if)#Ex
MLT-SW1(config)#
MLT-SW1(config)#
MLT-SW1(config)#
MLT-SW1(config)#Int vlan 60
MLT-SW1(config-if)#
MLT-SW1(config-if)#
MLT-SW1(config-if)#
MLT-SW1(config-if)#No sh
MLT-SW1(config-if)#
MLT-SW1(config-if)#Ip add 172.16.3.129 255.255.255.240
MLT-SW1(config-if)#
MLT-SW1(config-if)#Ip helper-address 172.16.3.130
MLT-SW1(config-if)#
MLT-SW1(config-if)#ex
```

```
MLT-SW2(config)#Int vlan 10
MLT-SW2(config-if)#
MLT-SW2(config-if)#
MLT-SW2(config-if)#
MLT-SW2(config-if)#No sh
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ip add 172.16.1.1 255.255.255.128
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ip helper-address 172.16.3.130
MLT-SW2(config-if)#
MLT-SW2(config-if)#ex
MLT-SW2(config)#
MLT-SW2(config)#
MLT-SW2(config)#
MLT-SW2(config)#Int vlan 20
MLT-SW2(config-if)#
MLT-SW2(config-if)#
MLT-SW2(config-if)#
MLT-SW2(config-if)#No sh
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ip add 172.16.1.129 255.255.255.128
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ip helper-address 172.16.3.130
MLT-SW2(config-if)#
MLT-SW2(config-if)#ex
MLT-SW2(config)#
MLT-SW2(config)#
MLT-SW2(config)#
MLT-SW2(config)#Int vlan 30
MLT-SW2(config-if)#
MLT-SW2(config-if)#
MLT-SW2(config-if)#
MLT-SW2(config-if)#No sh
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ip add 172.16.2.1 255.255.255.128
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ip helper-address 172.16.3.130
MLT-SW2(config-if)#
MLT-SW2(config-if)#ex
MLT-SW2(config)#
MLT-SW2(config)#
MLT-SW2(config)#
MLT-SW2(config)#Int vlan 40
MLT-SW2(config-if)#
MLT-SW2(config-if)#
MLT-SW2(config-if)#
MLT-SW2(config-if)#No sh
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ip add 172.16.2.129 255.255.255.128
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ip helper-address 172.16.3.130
MLT-SW2(config-if)#
MLT-SW2(config-if)#ex
MLT-SW2(config)#
MLT-SW2(config)#
MLT-SW2(config)#
MLT-SW2(config)#Int vlan 50
MLT-SW2(config-if)#
MLT-SW2(config-if)#
MLT-SW2(config-if)#
MLT-SW2(config-if)#No sh
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ip add 172.16.3.1 255.255.255.128
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ip helper-address 172.16.3.130
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ex
MLT-SW2(config)#
MLT-SW2(config)#
MLT-SW2(config)#
MLT-SW2(config)#Int vlan 60
MLT-SW2(config-if)#
MLT-SW2(config-if)#
MLT-SW2(config-if)#
MLT-SW2(config-if)#No sh
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ip add 172.16.3.129 255.255.255.240
MLT-SW2(config-if)#
MLT-SW2(config-if)#Ip helper-address 172.16.3.130
MLT-SW2(config-if)#
MLT-SW2(config-if)#ex
MLT-SW2(config)#
MLT-SW2(config)#Do wr
```

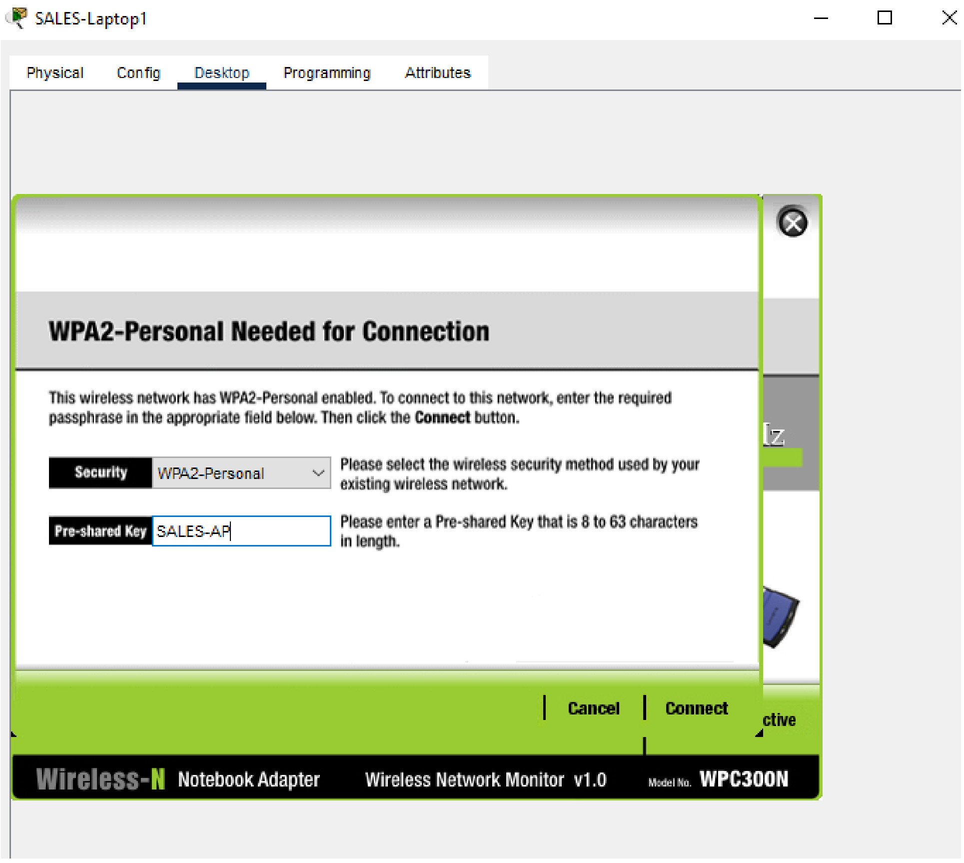
7. WIRLESS NETWORK CONFIG.



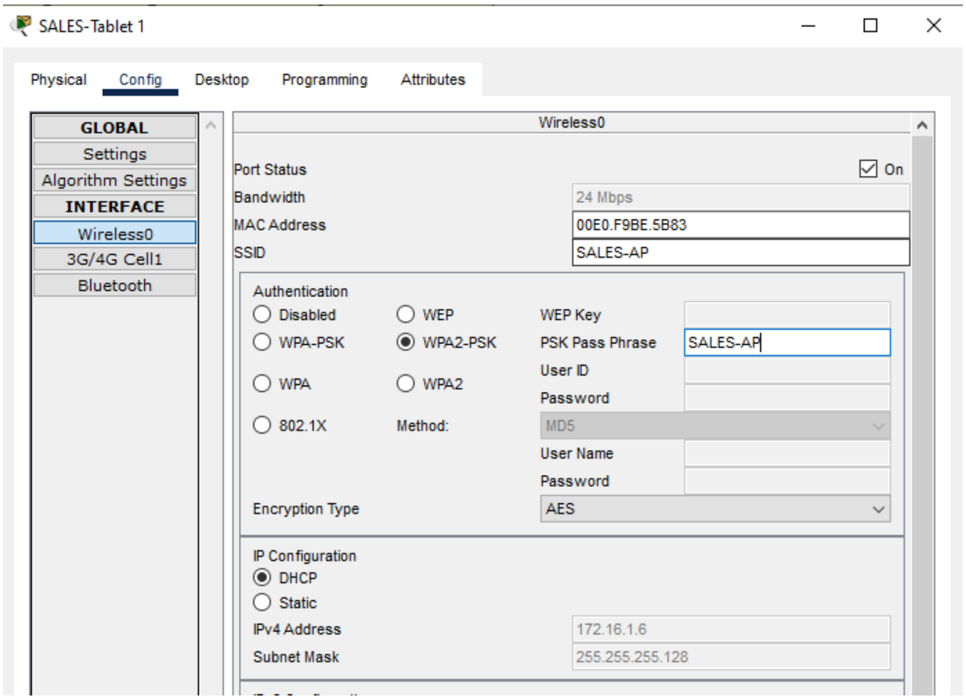

Off the Laptop and remove PT-MICROPHONE and place WPC300N and turn on the laptop.

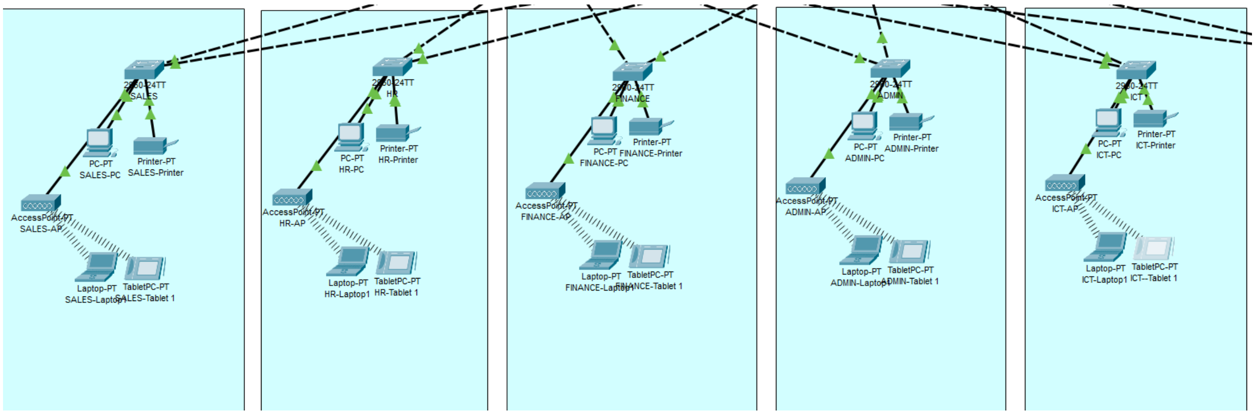



Desktop > PC wireless > CONNECT

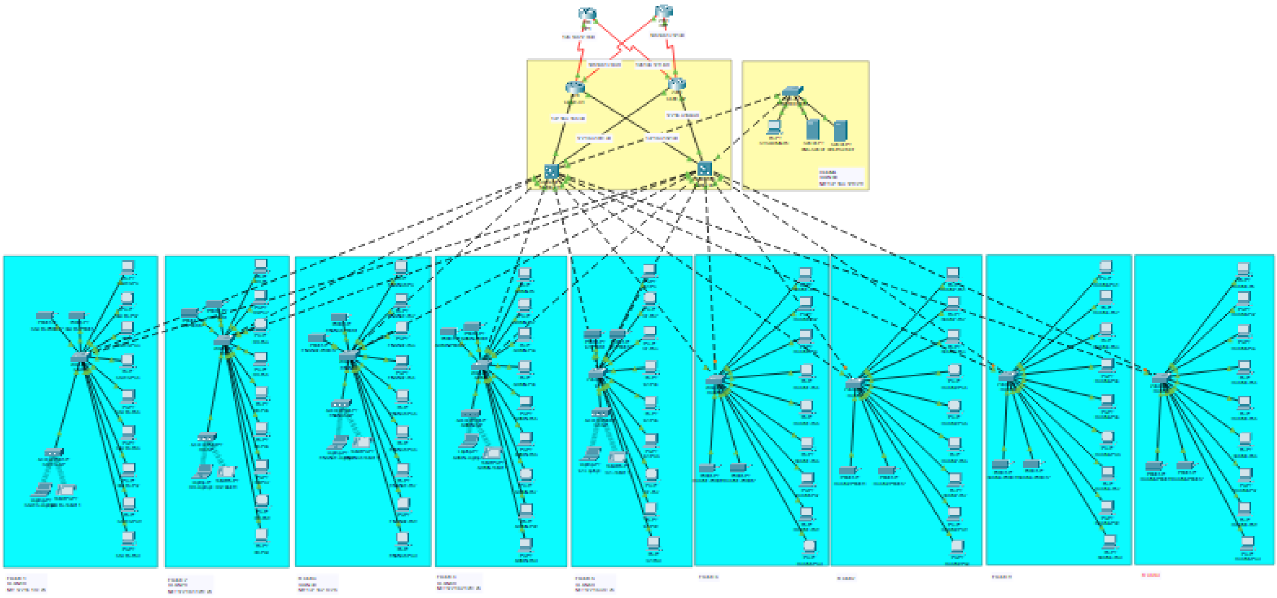



For the tablet






## 8. THE FINAL DESIGN.



# Conclusion

Network designers can offer valuable services to businesses by providing the network systems they use to perform their jobs properly. Working as a network designer can be an engaging career, full of opportunities.

- Problem-solving: Problem-solving skills can help you design effective networks capable of meeting businesses' needs. They can also help you diagnose network issues, install updates and recommend helpful system upgrades.
- Technical skills: Technical skills, like knowledge of programming languages and computer skills, are necessary for network designers. Employers often want candidates with strong practical knowledge and confidence in implementing network design concepts.
- Communication: Communication skills can help you interact with clients and company management to design successful network solutions. Interpersonal skills can also help you work with teammates and other networking professionals.
- Organizational skills: Organizational skills are also important for network designers because it helps them keep track of network needs like updates and security strategies.

# Reference


- Company Network Design & Implementation Using Cisco Packet Tracer | Enterprise Network Project #6
- https://www.auvik.com/franklyit/blog/network-design-best-practices/#:~:text=Network%20design%20is%20the%20practice%20of%20planning%20and,work%20to%20deploy%20and%20configure%20what%20was%20designed%29.
- https://www.indeed.com/career-advice/finding-a-job/how-to-become-network-designer#:~:text=Other%20skills%20to%20focus%20on%20include%3A%201%20Problem-solving%3A,to%20design%20successful%20network%20solutions.%20...%20More%20items