IT SECURITY PROJECT

Mina Atef Marzouk Gayed 22/05542

ABSTRACT

This project builds a secure network using Cisco Packet Tracer. It includes routers, switches, servers, an ASA firewall, and an SDN controller. The network uses OSPF routing, VPN, firewall rules, and AAA authentication to protect traffic and manage users. The goal is to design a stable and secure enterprise network.

SUPERVISE

Dr, Mostafa
Dr, Khaled

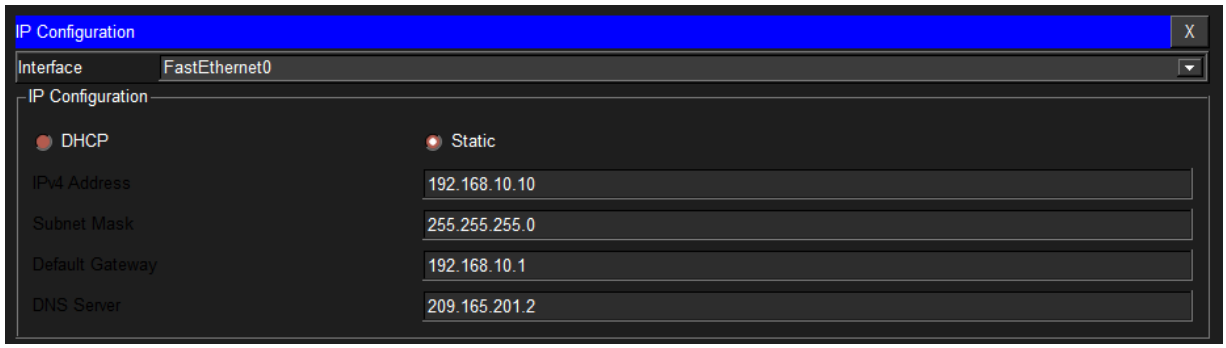
Mina Atef 22/05542

PART 1: Configure IP Address on All Devices

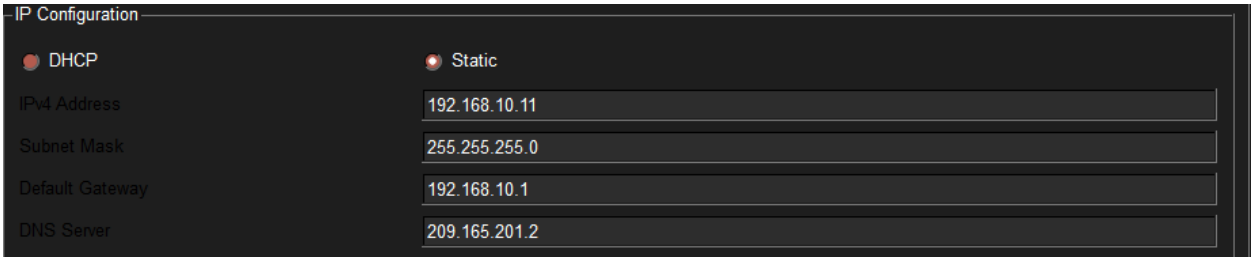
Configuration of ALL PCS

Device	Interface	IP Address / Prefix	Subnet Mask	DGW
PC 1	NIC	192.168.10.10/24	255.255.255.0	192.168.10.1
PC 2	NIC	192.168.10.11/24	255.255.255.0	192.168.10.1
PC 3	NIC	192.168.10.12/24	255.255.255.0	192.168.10.1
PC 4	NIC	192.168.20.20/24	255.255.255.0	192.168.20.1
PC 5	NIC	192.168.20.10/24	255.255.255.0	192.168.20.1
Guest PC	NIC	192.168.30.20/24	255.255.255.0	192.168.30.1
PC-B	NIC	192.168.50.10/24	255.255.255.0	192.168.50.1

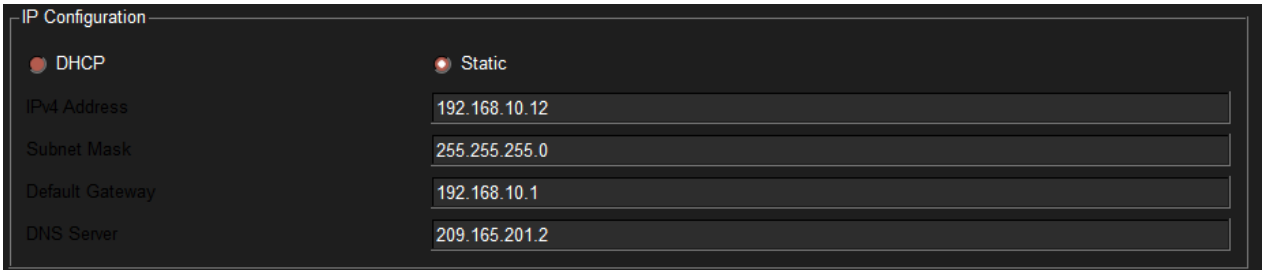
PC1



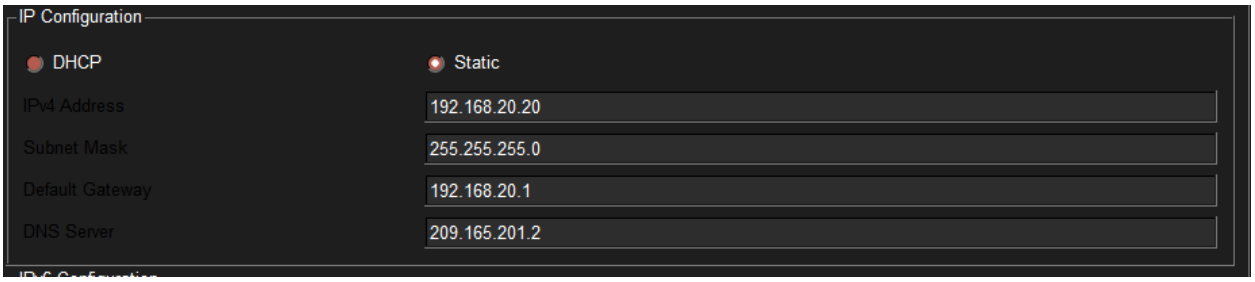
PC2



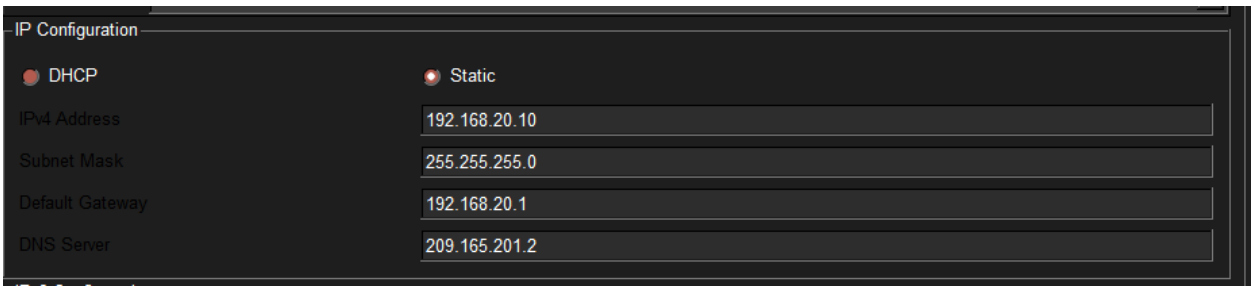
PC3



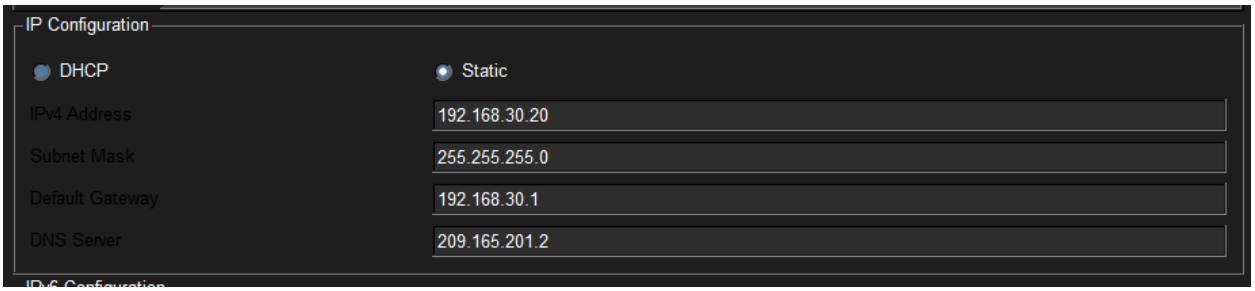
PC4



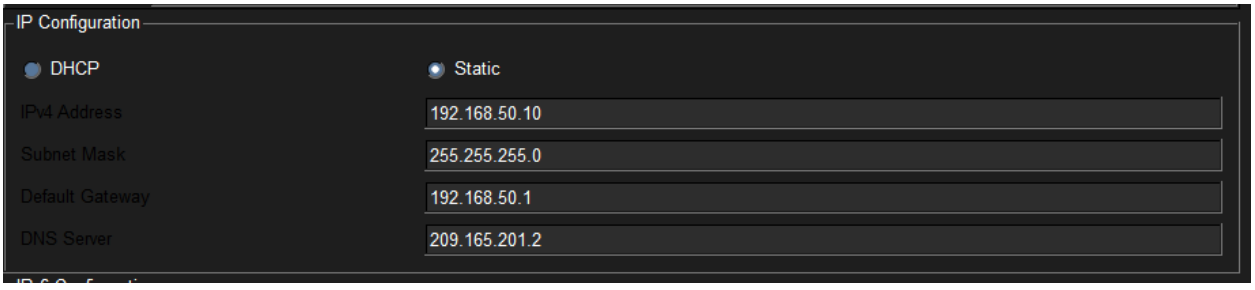
PC5



Guest PC



PC-B



Mina Atef 22/05542

Configuration of Routers

Device	Interface	IP Address / Prefix	Subnet Mask	DGW
R1	S0/1/0	10.1.1.1/30	255.255.255.252	N/A
	G0/0/1	192.168.10.1/24	255.255.255.0	N/A
ISP	S0/1/0	10.1.1.2/30	255.255.255.252	N/A
	S0/1/1	10.2.2.2/30	255.255.255.252	N/A
	G0/0/0	209.165.200.225/29	255.255.255.248	N/A
	G0/0/1	209.165.201.1/29	255.255.255.248	N/A
R2	S0/1/1	10.2.2.1/30	255.255.255.252	N/A
	G0/0/0	192.168.20.1/24	255.255.255.0	N/A
	G0/0/1	192.168.30.1/24	255.255.255.0	N/A

Show running-config

R1

```
interface GigabitEthernet0/0/1
 ip address 192.168.10.1 255.255.255.0
.
interface Serial0/1/0
 ip address 10.1.1.1 255.255.255.252
```

ISP

```
interface Serial0/1/0
 ip address 10.1.1.2 255.255.255.252
 ip ospf authentication key-chain MIU_SEC

interface Serial0/1/1
 ip address 10.2.2.2 255.255.255.252
:
interface GigabitEthernet0/0/0
 ip address 209.165.200.225 255.255.255.248
 duplex auto
 speed auto
!
interface GigabitEthernet0/0/1
 ip address 209.165.201.1 255.255.255.248
 duplex auto
 speed auto
```

R2

```
interface Serial0/1/1
 ip address 10.2.2.1 255.255.255.252
interface GigabitEthernet0/0/0
 ip address 192.168.20.1 255.255.255.0

interface GigabitEthernet0/0/1
 ip address 192.168.30.1 255.255.255.0
```

Configuration of Switches

Device	Interface	IP Address / Prefix	Proof /Subnet Mask	DGW
SW1	VLAN1	192.168.10.2/24	interface Vlan1 ip address 192.168.10.2 255.255.255.0	192.168.10.1
SW2	VLAN1	192.168.20.2/24	interface Vlan1 ip address 192.168.20.2 255.255.255.0	192.168.20.1
SW3	VLAN1	192.168.30.2/24	interface Vlan1 ip address 192.168.30.2 255.255.255.0	192.168.30.1
SW4	VLAN1	192.168.40.2/24	interface Vlan1 ip address 192.168.40.2 255.255.255.0	192.168.40.1
SW5	VLAN1	192.168.50.2/24	interface Vlan1 ip address 192.168.50.2 255.255.255.0	192.168.50.1
SW6	VLAN1	209.165.201.5/29	interface Vlan1 ip address 209.165.201.5 255.255.255.248	209.165.201.1

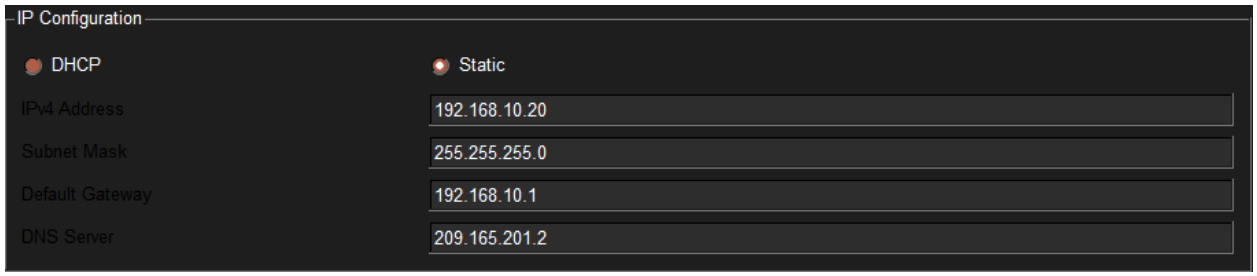
Configuration of ASA

Device	Interface	IP Address / Prefix	DGW
MIU-ASA	G1/1	209.165.200.226/29	N/A
	G1/2	192.168.50.1/24	N/A
	G1/3	192.168.40.1/24	N/A

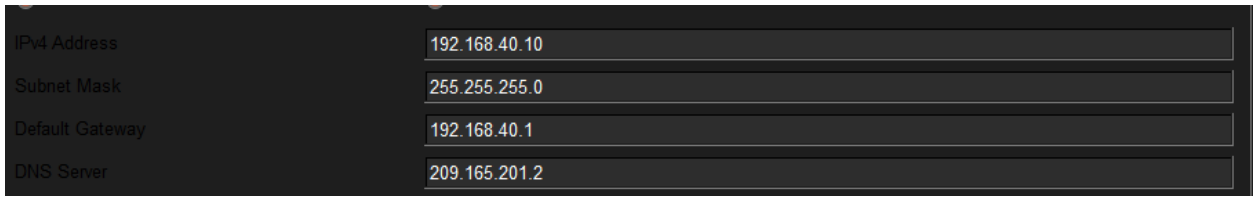
```
interface GigabitEthernet1/1
 nameif outside
 security-level 0
 ip address 209.165.200.226 255.255.255.248
!
interface GigabitEthernet1/2
 nameif inside
 security-level 100
 ip address 192.168.50.1 255.255.255.0
!
interface GigabitEthernet1/3
 nameif dmz
 security-level 60
 ip address 192.168.40.1 255.255.255.0
```

Syslog- NTP - AAA Server	NIC	192.168.10.20/24	192.168.10.1
DMZ _2_Server	NIC	192.168.40.10/24	192.168.40.1
DNS-Server	NIC	209.165.201.2/29	209.165.201.1
Miu.edu.eg	NIC	209.165.201.3/29	209.165.201.1
FTP and Email servers	NIC	209.165.201.4/29	209.165.201.1
PT-Controller0	NIC	192.168.10.30/24	192.168.10.1

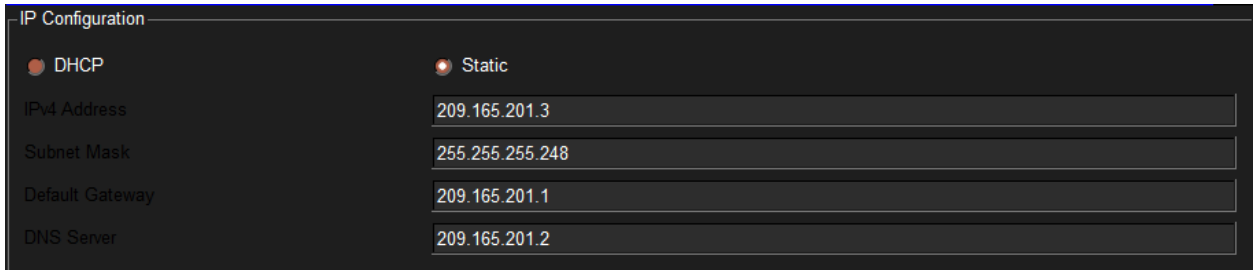
SYSLOG



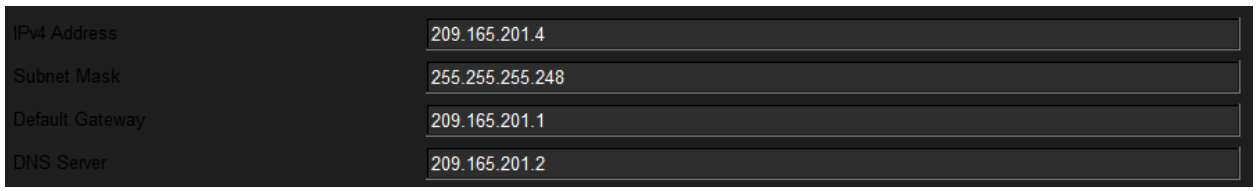
DMZ

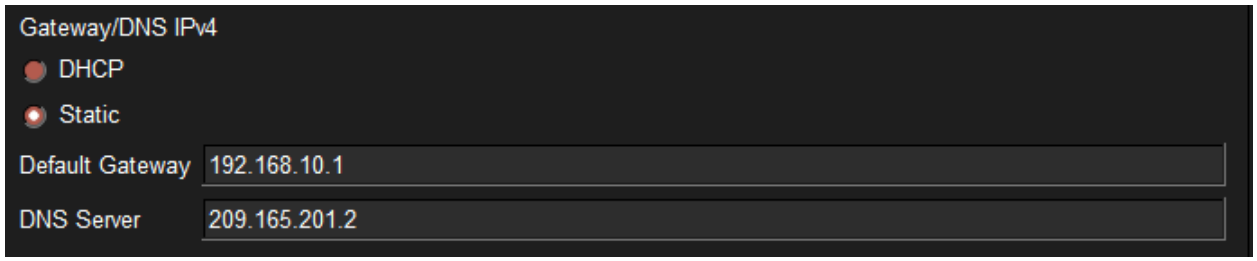


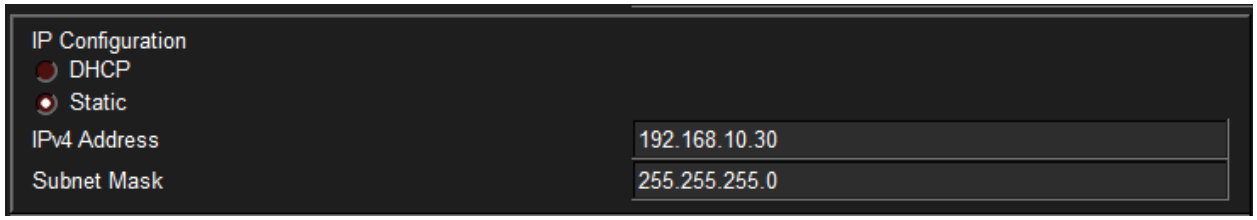
MIU.edu.eg



FTP







Summary

In Part 1, I set the IP addresses, subnet masks, and gateways for all devices. Each device now has the correct IP settings based on the table. This completes the basic IP setup.

****

Part 2: Basic Configuration and Configure SSH

Step 1: Configure Hostname

For each device, I need to set the hostname according to the addressing table.

```
SW-Ser(config)#hostname SW5
SW5(config)#
```

Step2: Configure Console and Enable Passwords

1st password for MIUpa55

2nd password class

```
! Configure console password
line console 0
password MIUpa55
login
exit

! Configure enable secret password
enable secret class
```

```
User Access Verification

Password:

SW6>en
SW6>enable
Password:
SW6#
```

Step3: Configure SSH on Configure

```
username IT-Sec1 privilege 15 secret miu123!
ip domain-name miu.edu.eg
crypto key generate rsa modulus 1024
!
line vty 0 15
transport input ssh
login local
exit
```

Verification that SSH is working fine

```
C:\>ssh -l IT-Sec1 192.168.10.2

Password:


SW1#
```

Summary : **I set the console and enable passwords, then configured SSH on all devices for secure remote access.**

Part 3: Configure OSPF Routing

Step 1: Configure OSPF on R1

router ospf 10
network 10.1.1.0 0.0.0.3 area 0
network 192.168.10.0 0.0.0.255 area 0

Step 2: Configure OSPF on ISP

router ospf 10
network 10.1.1.0 0.0.0.3 area 0
network 10.2.2.0 0.0.0.3 area 0
network 209.165.200.224 0.0.0.7 area 0
network 209.165.201.0 0.0.0.7 area 0
! Configure default route and propagate it
ip route 0.0.0.0 0.0.0.0 209.165.200.226
router ospf 10
default-information originate

Step 3: Configure OSPF on R2

router ospf 10
network 10.2.2.0 0.0.0.3 area 0
network 192.168.20.0 0.0.0.255 area 0
network 192.168.30.0 0.0.0.255 area 0

summary:I enabled OSPF on all routers so they can share their networks and learn routes from each other.

****

Part 4 Configure OSPF Routing Protocol Authentication by SHA256

Step 1: Configure a Key Chain on All Three Routers

Applied the above configuration on:R1,ISP,R2

```
key chain MIU_SEC
 key 10
  key-string MIU_AI_24
  cryptographic-algorithm hmac-sha-256
```

Step 2: Configure Serial Interfaces to Use the Key Chain

Applied the key chain to the serial interfaces that participate in OSPF adjacency.

R1

```
interface Serial0/1/0
 ip ospf authentication key-chain MIU_SEC
```

ISP

```
interface Serial0/1/0
 ip ospf authentication key-chain MIU_SEC

interface Serial0/1/1
 ip ospf authentication key-chain MIU_SEC
```

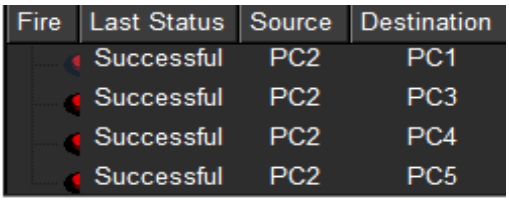
R2

```
interface Serial0/1/1
 ip ospf authentication key-chain MIU_SEC
```

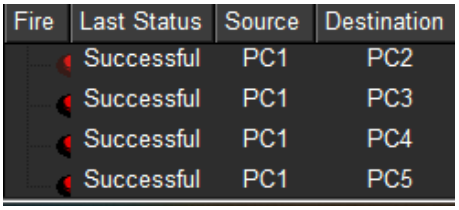
step 3: verification

- **show ip ospf interface**
- **show ip ospf neighbor**
- **show ip route**

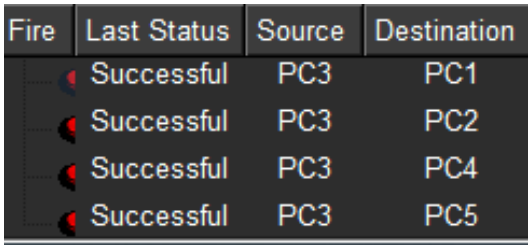
Used the ping command to verify connectivity between PC1, PC2, PC3, PC4 ,PC5.



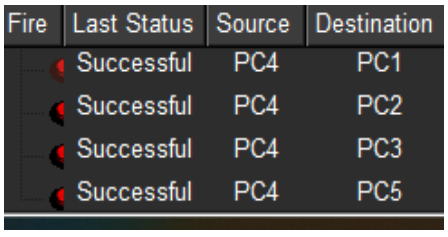
Fire	Last Status	Source	Destination
	Successful	PC2	PC1
	Successful	PC2	PC3
	Successful	PC2	PC4
	Successful	PC2	PC5



Fire	Last Status	Source	Destination
	Successful	PC1	PC2
	Successful	PC1	PC3
	Successful	PC1	PC4
	Successful	PC1	PC5



Fire	Last Status	Source	Destination
	Successful	PC3	PC1
	Successful	PC3	PC2
	Successful	PC3	PC4
	Successful	PC3	PC5

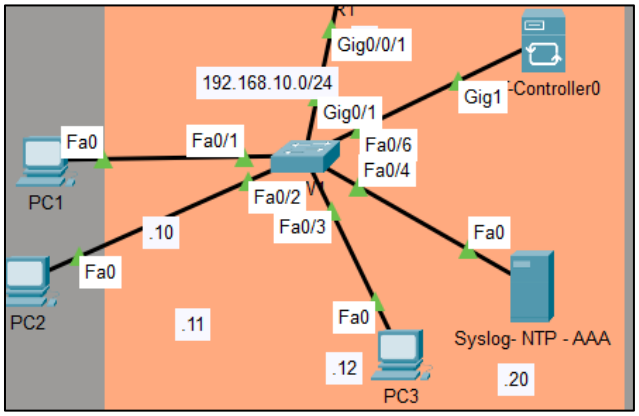


Fire	Last Status	Source	Destination
	Successful	PC4	PC1
	Successful	PC4	PC2
	Successful	PC4	PC3
	Successful	PC4	PC5

summary :**I added SHA-256 authentication to OSPF so only trusted routers can form OSPF neighbors.**

Part 5: Switch Security Configuration

According to image



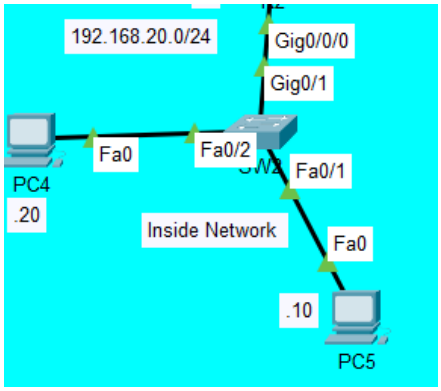
SW1 – Active Ports

- **Fa0/1 → PC1**
- **Fa0/2 → PC2**
- **Fa0/3 → PC3**
- **Fa0/4 → Syslog–NTP–AAA**
- **Fa0/6 → PT-Controller**
- **Gi0/1 → R1**
  Unused: **Fa0/5, Fa0/7–Fa0/24, Gi0/2**

```
SW1(config)# interface range Fa0/5 , Fa0/7 - Fa0/24 , Gi0/2
SW1(config-if-range)# shutdown
vlan 888
 name BlackHole
SW1(config)# interface range Fa0/5 , Fa0/7 - Fa0/24 , Gi0/2
SW1(config-if-range)# switchport mode access
SW1(config-if-range)# switchport access vlan 888
```

SW2 – Active Ports

- **Fa0/1 → PC5**
- **Fa0/2 → PC4**
- **Gi0/1 → R2**
  Unused: **Fa0/3–Fa0/24, Gi0/2**




```
SW2(config)# interface range Fa0/3 - Fa0/24 , Gi0/2
SW2(config-if-range)# shutdown
vlan 888
 name BlackHole
SW2(config)# interface range Fa0/3 - Fa0/24 , Gi0/2
SW2(config-if-range)# switchport mode access
SW2(config-if-range)# switchport access vlan 888
```

Summary: I shut down all unused ports on SW1 and SW2 and moved them to VLAN 888 (BlackHole).

Step 1: Implement Port Security on SW1 and SW2

Mode	Port Action	Traffic Behavior	Logs
shutdown	Port turns off (err-disabled)	Drops violating traffic	Yes
restrict	Port stays up	Drops violating traffic	**Yes**
protect	Port stays up	Drops violating traffic	No

Configure the port security violation mode to **drop packets** from MAC addresses that exceed the maximum, generate **a Syslog entry**, but **not disable the ports**.

So I will use restricted

Apply port security only on active access ports**.**

SW1 – Active Access Ports:

```
interface range Fa0/1 - Fa0/4,Fa0/6
SW1(config-if-range)# switchport mode access
SW1(config-if-range)# switchport port-security
SW1(config-if-range)# switchport port-security maximum 5
SW1(config-if-range)# switchport port-security mac-address sticky
SW1(config-if-range)# switchport port-security violation restrict
```

SW2 – Active Access Ports:

Fa0/1 (PC5), Fa0/2 (PC4)

```
interface range Fa0/1 , Fa0/2
SW2(config-if-range)# switchport mode access
SW2(config-if-range)# switchport port-security
SW2(config-if-range)# switchport port-security maximum 5
SW2(config-if-range)# switchport port-security mac-address sticky
SW2(config-if-range)# switchport port-security violation restrict
```

Step 2: Configure PortFast and BPDU Guard

a. Enable PortFast on all the access ports that are in use on SW-2

```
SW2(config)# interface range Fa0/1 , Fa0/2
SW2(config-if-range)# spanning-tree portfast
```

b. Enable BPDU Guard on all the access ports that are in use on SW-2

```
SW2(config)# interface range Fa0/1 , Fa0/2
SW2(config-if-range)# spanning-tree bpduguard enable
```

Step 3: Configure DHCP Snooping on SW-1

a. Configure the router and server ports as trusted

Enable DHCP snooping:

```
SW1(config)# ip dhcp snooping
SW1(config)# ip dhcp snooping vlan 1
```

Mark trusted ports:

```
SW1(config)# interface Gi0/1
SW1(config-if)# ip dhcp snooping trust

SW1(config)# interface Fa0/4
SW1(config-if)# ip dhcp snooping trust
```

b. Limit untrusted ports to 5 DHCP packets per second

Untrusted access ports on SW1:

- **Fa0/1-Pc1**
- **Fa0/2-PC2**
- **Fa0/3-PC3**
- **Fa0/6-PTcontroller**

Apply limit:

```
SW1(config)# interface range Fa0/1 , Fa0/2 , Fa0/3 , Fa0/6
SW1(config-if-range)# ip dhcp snooping limit rate 5
```

→This means the port is only allowed to send 5 DHCP packets per second.
If it sends more, the switch blocks it.

Summary: I added port security, enabled PortFast and BPDU Guard, and configured DHCP snooping to protect the switches from unsafe devices and attacks.

****

PART 6 :Site-to-Site IPsec VPN Configuration (R1 ↔ R2)

ISAKMP Phase 1 Parameters (R1 & R2)

- **Method: ISAKMP**
- **Encryption: AES-256**
- **Hash: SHA-1**
- **Authentication: pre-shared-key**
- **DH Group: 5**
- **Lifetime: 86400**
- **Shared key:** VPN_miupa55

IPsec Phase 2 Parameters (R1 & R2)

- **Transform set:** VPN-SET
- **Encryption: esp-aes**
- **Authentication: esp-sha-hmac**
- **Crypto map name:** VPN-MAP
- **Mode: ipsec-isakmp**
- **Interesting traffic:**
  - **R1: ACL 110 → src 192.168.10.0 dst 192.168.20.0**
  - **R2: ACL 110 → src 192.168.20.0 dst 192.168.10.0**

CONFIGURE VPN ON R1

Step 1: Test regular routing

Fire	Last Status	Source	Destination	Type
	Successful	PC1	PC5	ICMP

Step 2: Enable Security License

```
R1(config)# license boot level securityk9
R1# write memory
R1# reload
```

Verify:

```
R1# show version
securityk9       securityk9       Permanent        securityk9
```

Step 3: Define Interesting Traffic (ACL 110)

```
R1(config)# access-list 110 permit ip 192.168.10.0 0.0.0.255 192.168.20.0
0.0.0.255
```

Step 4: Configure IKE Phase 1 (ISAKMP Policy)

```
R1(config)# crypto isakmp policy 20
R1(config-isakmp)# encryption aes 256
R1(config-isakmp)# hash sha
R1(config-isakmp)# authentication pre-share
R1(config-isakmp)# group 5
R1(config-isakmp)# lifetime 86400
```

Configure shared key:

```
R1(config)# crypto isakmp key VPN_miupa55 address 10.2.2.1→R2
```

Step 5: Configure IKE Phase 2 (IPsec)

a. Create transform set

```
R1(config)# crypto ipsec transform-set VPN-SET esp-aes esp-sha-hmac
```

b. Create crypto map

```
R1(config)# crypto map VPN-MAP 10 ipsec-isakmp
R1(config-crypto-map)# set peer 10.2.2.1→R2
R1(config-crypto-map)# set transform-set VPN-SET
R1(config-crypto-map)# match address 110
```

Step 6: Apply Crypto Map to Serial Interface

```
R1(config)# interface Serial0/1/0
R1(config-if)# crypto map VPN-MAP
```

The same VPN configuration was applied on R2, with the ACL source and destination reversed, using Serial0/1/1 and peer IP address 10.1.1.1.

Verify that is working

```
R1#show crypto MAP
Crypto Map VPN-MAP 10 ipsec-isakmp
        Peer = 10.2.2.1
        Extended IP access list 110
            access-list 110 permit ip 192.168.10.0 0.0.0.255 192.168.20.0 0.0.0.255
        Current peer: 10.2.2.1
        Security association lifetime: 4608000 kilobytes/3600 seconds
        PFS (Y/N): N
        Transform sets={
                VPN-SET,
        }
        Interfaces using crypto map VPN-MAP:
                Serial0/1/0
```

```
R1#SHOW crypto ipsec sa

interface: Serial0/1/0
    Crypto map tag: VPN-MAP, local addr 10.1.1.1

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (192.168.10.0/255.255.255.0/0/0)
   remote  ident (addr/mask/prot/port): (192.168.20.0/255.255.255.0/0/0)
   current_peer 10.2.2.1 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 9, #pkts encrypt: 9, #pkts digest: 0
   #pkts decaps: 7, #pkts decrypt: 7, #pkts verify: 0
   #pkts compressed: 0, #pkts decompressed: 0
   #pkts not compressed: 0, #pkts compr. failed: 0
   #pkts not decompressed: 0, #pkts decompress failed: 0
   #send errors 1, #recv errors 0

     local crypto endpt.: 10.1.1.1, remote crypto endpt.:10.2.2.1
     path mtu 1500, ip mtu 1500, ip mtu idb Serial0/1/0
     current outbound spi: 0x231BF4E9(589034729)

     inbound esp sas:
      spi: 0x09E94B0C(166284044)
        transform: esp-aes esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 2009, flow_id: FPGA:1, crypto map: VPN-MAP
        sa timing: remaining key lifetime (k/sec): (4525504/3408)
        IV size: 16 bytes
        replay detection support: N
        Status: ACTIVE

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0x231BF4E9(589034729)
        transform: esp-aes esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 2010, flow_id: FPGA:1, crypto map: VPN-MAP
        sa timing: remaining key lifetime (k/sec): (4525504/3408)
        IV size: 16 bytes
        replay detection support: N
        Status: ACTIVE

     outbound ah sas:

     outbound pcp sas:
```

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Successful	PC1	PC4	ICMP		0.000	N	0	(edit)	(delete)
	Successful	PC1	PC5	ICMP		0.000	N	1	(edit)	(delete)

Its working fine as it encrypts the packets and send and receives packet successfully

VPN Summary: I configured and verified a site-to-site IPsec VPN between R1 and R2 to securely encrypt traffic between their LANs.

PART 7 – ZONE-BASED POLICY FIREWALL ON R2

1) INSIDE → OUTSIDE

1. Create zones

```
conf t
zone security INSIDE
zone security OUTSIDE
```

Create the security zones.	Inside zone name: **INSIDE** Outside zone name: **OUTSIDE**

2. Create Class-map

```
class-map type inspect match-any INSIDE-PROTOCOLS
 match protocol tcp
 match protocol udp
 match protocol icmp
```

Create an inspect class map.	Class map name: **INSIDE-PROTOCOLS** Inspection type: **match-any** Protocols allowed: **tcp,udp,icmp**

3. Create Policy-map and bind it with class map

```
policy-map type inspect INSIDE-TO-OUTSIDE-PM
 class type inspect INSIDE-PROTOCOLS
  inspect
```

Create an inspect policy map.	Policy map name: **INSIDE-TO-OUTSIDE-PM** Bind the class map to the policy map. Matched packets should be inspected.

4. Create Zone Pair & Apply the policy map to the zone pair.

```
zone-pair security INSIDE-TO-OUTSIDE-ZP source INSIDE destination OUTSIDE
service-policy type inspect INSIDE-TO-OUTSIDE-PM
```

Create a zone pair.	Zone pair name: **INSIDE-TO-OUTSIDE-ZP** Source zone: **INSIDE** Destination zone: **OUTSIDE**
Apply the policy map to the zone pair.	Zone pair name: **INSIDE-TO-OUTSIDE-ZP** Policy map name: **INSIDE-TO-OUTSIDE-PM**

5. Assign interfaces

```
interface g0/0/0
 zone-member security INSIDE

interface s0/1/1
 zone-member security OUTSIDE
```

Assign interfaces to the proper security zones.	Interface G0/0/0: **INSIDE** Interface S0/1/1: **OUTSIDE**

2) GUEST → OUTSIDE (Only HTTP/HTTPS/DNS)

1. Create zone

```
zone security GUEST
```

Create the security zone.	GUEST zone name: **GUEST**

2. Class-map

```
class-map type inspect match-any GUEST-PROTOCOLS
 match protocol http
 match protocol https
 match protocol dns
```

Create an inspect class map.	Class map name: **GUEST-PROTOCOLS** Inspection type: **match-any** Protocols allowed: **http,https.dns**

3. Policy-map and bind it with class map

```
policy-map type inspects GUEST-TO-OUTSIDE-PM
 class type inspect GUEST-PROTOCOLS
  inspect
```

Create an inspect policy map.	Policy map name: **GUEST-TO-OUTSIDE-PM** Bind the class map to the policy map. Matched packets should be inspected.

4. Zone pair & Apply policy map to zone pair

```
zone-pair security GUEST-TO-OUTSIDE-ZP source GUEST destination OUTSIDE
 service-policy type inspect GUEST-TO-OUTSIDE-PM
```

Create a zone pair.	Zone pair name: **GUEST-TO-OUTSIDE-ZP** Source zone: **GUEST** Destination zone: **OUTSIDE**
Apply the policy map to the zone pair.	Zone pair name: **GUEST-TO-OUTSIDE-ZP** Policy map name: **GUEST-TO-OUTSIDE-PM**

5. Assign interface

Guest network is G0/0/1:

```
interface g0/0/1
 zone-member security GUEST
```

Assign interfaces to the proper security zones.	Interface G0/0/1: **GUEST**

3)OUTSIDE → INSIDE (Allow VPN traffic ONLY)

1. ACL for VPN-allowed traffic

```
`ip access-list extended REMOTE-TRAFFIC
 permit ip 192.168.10.0 0.0.0.255 192.168.20.0 0.0.0.255
```

Create a named ACL to allow R1 VPN traffic through to **192.168.20.0 /24**	Name: **REMOTE-TRAFFIC** Source: **192.168.10.0 /24** Destination: **192.168.20.0 /24**

2. Class-map

```
class-map type inspect match-all OUTSIDE-TRAFFIC
 match access-group name REMOTE-TRAFFIC
```

Create an inspect class map.	Class map name: **OUTSIDE-TRAFFIC** Inspection type: **match-all** Access group allowed: **REMOTE-TRAFFIC**

3. Policy map and bind it with class map

```
policy-map type inspect OUTSIDE-TO-INSIDE-PM
 class type inspect OUTSIDE-TRAFFIC
  inspect
```

Create an inspect policy map.	Policy map name: **OUTSIDE-TO-INSIDE-PM** Bind the class map to the policy map. Matched packets should be inspected.

4. Zone pair bind it with policy map

```
zone-pair security OUTSIDE-TO-INSIDE-ZP source OUTSIDE destination INSIDE
 service-policy type inspect OUTSIDE-TO-INSIDE-PM
```

Create a zone pair.	Zone pair name: **OUTSIDE-TO-INSIDE-ZP** Source zone: **OUTSIDE** Destination zone: **INSIDE**
Apply the policy map to the zone pair.	Zone pair name: **OUTSIDE-TO-INSIDE-ZP** Policy map name: **OUTSIDE-TO-INSIDE-PM**

4) OUTSIDE → GUEST (Allow VPN traffic ONLY)

1. ACL

```
ip access-list extended REMOTE-TRAFFIC_G
 permit ip 192.168.10.0 0.0.0.255 192.168.30.0 0.0.0.255
```

Create a named ACL to allow R1 VPN traffic through to **192.168.30.0 /24**	Name: **REMOTE-TRAFFIC_G** Source: **192.168.10.0 /24** Destination: **192.168.30.0 /24**

2. Class-map

```
class-map type inspect match-all OUTSIDE-TRAFFIC_G
 match access-group name REMOTE-TRAFFIC_G
```

Create an inspect class map.	Class map name: **OUTSIDE-TRAFFIC_G** Inspection type: **match-all** Access group allowed: **REMOTE-TRAFFIC_G**

3. Policy map bind it with class-Map

```
policy-map type inspect OUTSIDE-TO-GUEST-PM
 class type inspect OUTSIDE-TRAFFIC_G
  inspect
```

Create an inspect policy map.	Policy map name: **OUTSIDE-TO-GUEST-PM** Bind the class map to the policy map. Matched packets should be inspected.

4. Zone pair bind it with policy map

```
zone-pair security OUTSIDE-TO-GUEST-ZP source OUTSIDE destination GUEST
 service-policy type inspect OUTSIDE-TO-GUEST-PM
```

Create a zone pair.	Zone pair name: **OUTSIDE-TO-INSIDE-ZP** Source zone: **OUTSIDE** Destination zone: **GUEST**
Apply the policy map to the zone pair.	Zone pair name: **OUTSIDE-TO- GUEST -ZP** Policy map name: **OUTSIDE-TO- GUEST -PM**

STEP 5 – VERIFICATION

1. Test Internet access

From PC4, PC5, Guest PC:

```
ping miu.edu.eg
```

```
C:\>ping miu.edu.eg

Pinging 209.165.201.3 with 32 bytes of data:

Reply from 209.165.201.3: bytes=32 time=2ms TTL=126
Reply from 209.165.201.3: bytes=32 time=11ms TTL=126
Reply from 209.165.201.3: bytes=32 time=1ms TTL=126
Reply from 209.165.201.3: bytes=32 time=1ms TTL=126

Ping statistics for 209.165.201.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 11ms, Average = 3ms
```

2. Test VPN

PC1 → PC5:

```
ping 192.168.20.10
```

```
C:\>ping 192.168.20.10

Pinging 192.168.20.10 with 32 bytes of data:

Reply from 192.168.20.10: bytes=32 time=2ms TTL=126
Reply from 192.168.20.10: bytes=32 time=2ms TTL=126
Reply from 192.168.20.10: bytes=32 time=15ms TTL=126
Reply from 192.168.20.10: bytes=32 time=2ms TTL=126

Ping statistics for 192.168.20.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 15ms, Average = 5ms
```

Check VPN:

```
show crypto isakmp sa
show crypto ipsec sa
```

```
R1#show  crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id slot status
10.2.2.1        10.1.1.1        QM IDLE           1008    0 ACTIVE
```

```
   #pkts encaps: 15, #pkts encrypt: 15, #pkts digest: 0
   #pkts decaps: 13, #pkts decrypt: 13, #pkts verify: 0
   #pkts compressed: 0, #pkts decompressed: 0
   #pkts not compressed: 0, #pkts compr. failed: 0
   #pkts not decompressed: 0, #pkts decompress failed: 0
   #send errors 1, #recv errors 0
```

3. Verify ZPF

show zone security

```
R2#show zone security
zone self
  Description: System defined zone


zone INSIDE
  Member Interfaces:
    GigabitEthernet0/0/0


zone OUTSIDE
  Member Interfaces:
    Serial0/1/1


zone GUEST
  Member Interfaces:
    GigabitEthernet0/0/1
```

show zone-pair security

```
R2#show zone-pair security
Zone-pair name OUTSIDE-TO-INSIDE-ZP
    Source-Zone OUTSIDE  Destination-Zone INSIDE
    service-policy OUTSIDE-TO-INSIDE-PM

Zone-pair name INSIDE-TO-OUTSIDE-ZP
    Source-Zone INSIDE  Destination-Zone OUTSIDE
    service-policy INSIDE-TO-OUTSIDE-PM

Zone-pair name GUEST-TO-OUTSIDE-ZP
    Source-Zone GUEST  Destination-Zone OUTSIDE
    service-policy GUEST-TO-OUTSIDE-PM

Zone-pair name OUTSIDE-TO-GUEST-ZP
    Source-Zone OUTSIDE  Destination-Zone GUEST
    service-policy OUTSIDE-TO-GUEST-PM
```

Outside cant reach inside but vice verse can

Fire	Last Status	Source	Destination	Type	Color
	Failed	ISP	PC4	ICMP	
	Failed	ISP	PC5	ICMP	
	Successful	PC5	ISP	ICMP	
	Successful	PC4	ISP	ICMP	

Part 7 Summary: I used a zone-based policy firewall on R2 to allow full access from INSIDE, limited web access from GUEST, and block OUTSIDE traffic except VPN.

Part 8: Configure a stateless firewall on R1 using a named extended ACL

- Block **HTTP** and **HTTPS** from **PC1** → **Miu.edu.eg**
- Block **FTP** (TCP port 21) from **PC2** → **FTP** Server
- Block **ICMP** from **PC3** → **Miu.edu.eg + FTP Server**

Create named extended ACL BLOCK_TRAFFIC

```
ip access-list extended BLOCK_TRAFFIC
```

Deny PC1 HTTP/HTTPS to Miu.edu.eg

```
deny tcp host 192.168.10.10 host 209.165.201.3 eq 80
deny tcp host 192.168.10.10 host 209.165.201.3 eq 443
```

Deny PC2 FTP (port 21) to FTP Server

```
deny tcp host 192.168.10.11 host 209.165.201.4 eq 21
```

Deny PC3 ICMP to Miu.edu.eg & ICMP to FTP Server

```
deny icmp host 192.168.10.12 host 209.165.201.3
deny icmp host 192.168.10.12 host 209.165.201.4
```

8. Permit all remaining traffic

```
permit ip any any
exit
```

verification

```
R1#show ip access-lists BLOCK_TRAFFIC
Extended IP access list BLOCK_TRAFFIC
    deny tcp host 192.168.10.10 host 209.165.201.3 eq www (39 match(es))
    deny tcp host 192.168.10.10 host 209.165.201.3 eq 443 (188 match(es))
    deny tcp host 192.168.10.11 host 209.165.201.4 eq ftp (36 match(es))
    deny tcp host 192.168.10.12 host 209.165.201.3
    deny tcp host 192.168.10.12 host 209.165.201.4
    deny icmp host 192.168.10.12 host 209.165.201.3 (4 match(es))
    deny icmp host 192.168.10.12 host 209.165.201.4 (4 match(es))
    permit ip any any (10 match(es))
```

Part 8.2: Apply and Verify the Extended ACL

1. Apply the ACL to the correct interface

Commands

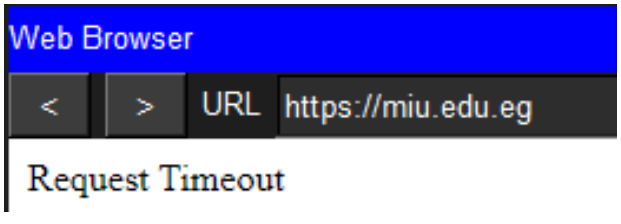
```
R1(config)# interface g0/0/1
R1(config-if)# ip access-group BLOCK_TRAFFIC in
R1(config-if)# exit
```

---

2. Verification Steps

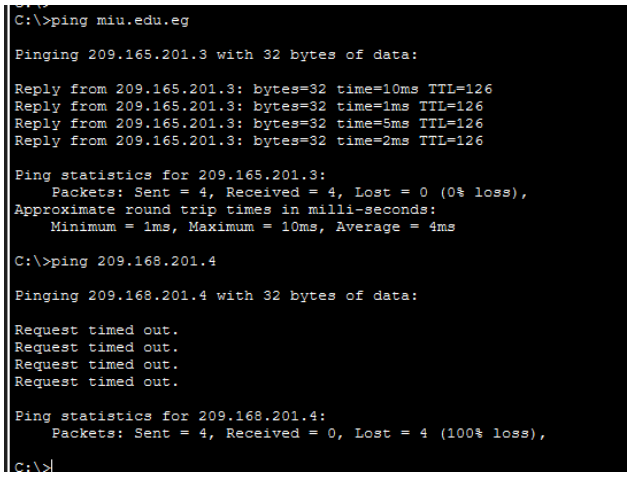
PC1 Tests

HTTP/HTTPS to miu.edu.eg



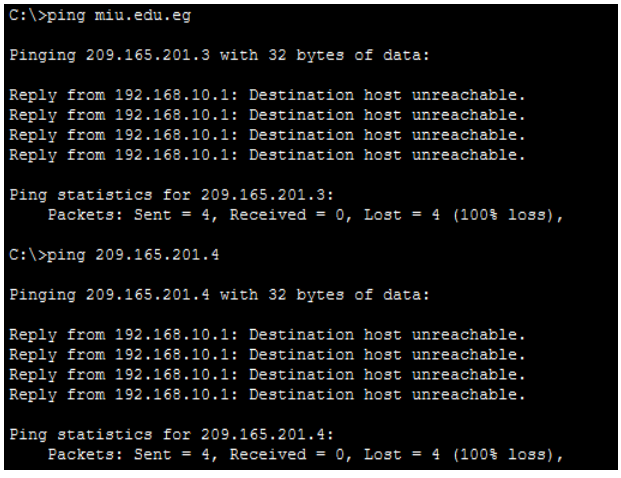
PC2 Tests

FTP to FTP server



PC3 Tests

ICMP blocks



Part 9 – ASA Basic Settings and Firewall Using CLI

Step 1: Set hostname and domain

```
configure terminal
hostname MIU-ASA
domain-name miu.com
```

Step 2: Set date and time

```
clock set 15:10:00 23 Nov 2025
```

Step 3: Configure OUTSIDE and INSIDE interfaces

a. OUTSIDE → Gi1/1

```
interface g1/1
nameif OUTSIDE
ip address 209.165.200.226 255.255.255.248
security-level 0
no shutdown
exit
```

b. INSIDE → Gi1/2

```
interface g1/2
nameif INSIDE
ip address 192.168.50.1 255.255.255.0
security-level 100
no shutdown
exit
```

Step 3c: Verify interfaces

1) Interface status

```
show interface ip brief
```

```
MIU-ASA#show interface ip brief
Interface              IP-Address      OK? Method Status                Protocol
Virtual0               127.1.0.1       YES unset  up                    up
GigabitEthernet1/1     209.165.200.226 YES manual up                    up
GigabitEthernet1/2     192.168.50.1    YES manual up                    up
```

2) IP information

```
show ip address
```

```
MIU-ASA#show ip address
System IP Addresses:
Interface              Name                IP address      Subnet mask     Method
GigabitEthernet1/1     OUTSIDE             209.165.200.226 255.255.255.248 manual
GigabitEthernet1/2     INSIDE              192.168.50.1    255.255.255.0   manual
GigabitEthernet1/3     dmz                 192.168.40.1    255.255.255.0   manual
GigabitEthernet1/4                         unassigned      unassigned      unset
GigabitEthernet1/5                         unassigned      unassigned      unset
GigabitEthernet1/6                         unassigned      unassigned      unset
GigabitEthernet1/7                         unassigned      unassigned      unset
GigabitEthernet1/8                         unassigned      unassigned      unset
Management1/1                              unassigned      unassigned      unset

Current IP Addresses:
Interface              Name                IP address      Subnet mask     Method
GigabitEthernet1/1     OUTSIDE             209.165.200.226 255.255.255.248 manual
GigabitEthernet1/2     INSIDE              192.168.50.1    255.255.255.0   manual
GigabitEthernet1/3     dmz                 192.168.40.1    255.255.255.0   manual
GigabitEthernet1/4                         unassigned      unassigned      unset
GigabitEthernet1/5                         unassigned      unassigned      unset
GigabitEthernet1/6                         unassigned      unassigned      unset
GigabitEthernet1/7                         unassigned      unassigned      unset
GigabitEthernet1/8                         unassigned      unassigned      unset
Management1/1                              unassigned      unassigned      unset
```

Step 4: Connectivity Tests

a. From PC-B → ASA INSIDE

PC-B IP: 192.168.50.10
ASA inside IP: 192.168.50.1

On PC-B:

```
ping 192.168.50.1
```

```
Pinging 192.168.50.1 with 32 bytes of data:

Reply from 192.168.50.1: bytes=32 time<1ms TTL=255
Reply from 192.168.50.1: bytes=32 time<1ms TTL=255
Reply from 192.168.50.1: bytes=32 time<1ms TTL=255
Reply from 192.168.50.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.50.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

b. From PC-B → ASA OUTSIDE

ASA outside IP: 209.165.200.226

On PC-B:

```
ping 209.165.200.226
```

```
C:\>ping 209.165.200.226

Pinging 209.165.200.226 with 32 bytes of data:

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

Ping statistics for 209.165.200.226:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Step 1: Configure the ASA Default Route

a. Add default route (quad zero)

The ASA must send all unknown traffic to the ISP router at 209.165.200.225 through interface OUTSIDE.

Command:

```
MIU-ASA(config)# route OUTSIDE 0.0.0.0 0.0.0.0 209.165.200.225
```

b. Verify the route

```
MIU-ASA# show route

S*   0.0.0.0/0 [1/0] via 209.165.200.225
```

Step 2: Configure PAT Using Network Objects

a. Create the object

INSIDE network is 192.168.50.0/24
Create the inside object:

```
MIU-ASA(config)# object network INSIDE-NET
MIU-ASA(config-network-object)# subnet 192.168.50.0 255.255.255.0
MIU-ASA(config-network-object)# nat (INSIDE,OUTSIDE) dynamic interface
MIU-ASA(config-network-object)# exit
```

This enables PAT using the outside interface IP 209.165.200.226.

b. Verify NAT object

```
MIU-ASA# show run object
MIU-ASA#show nat
Auto NAT Policies (Section 2)
1 (DMZ) to (OUTSIDE) source static DMZ-SERVER 209.165.200.227
    translate_hits = 0, untranslate_hits = 0
2 (INSIDE) to (OUTSIDE) source dynamic INSIDE-NET interface
    translate_hits = 5, untranslate_hits = 3

S*   0.0.0.0/0 [1/0] via 209.165.200.225
MIU-ASA#
```

this is pc b

a. Display the current inspection configuration

```
show run policy-map

policy-map global_policy
 class inspection_default
  inspect dns preset_dns_map
  inspect ftp
  inspect tftp
```

---

b. Add ICMP inspection to the global policy

```
configure terminal
policy-map global_policy
 class inspection_default
  inspect icmp
exit
```

---

c. Verify ICMP inspection is active

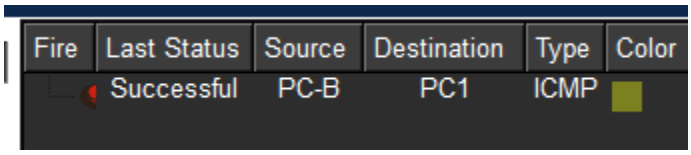
```
show run policy-map

Output

policy-map global_policy
 class inspection_default
  inspect dns preset_dns_map
  inspect ftp
  inspect icmp
  inspect tftp
```

---

d. Test from PC-B to PC1

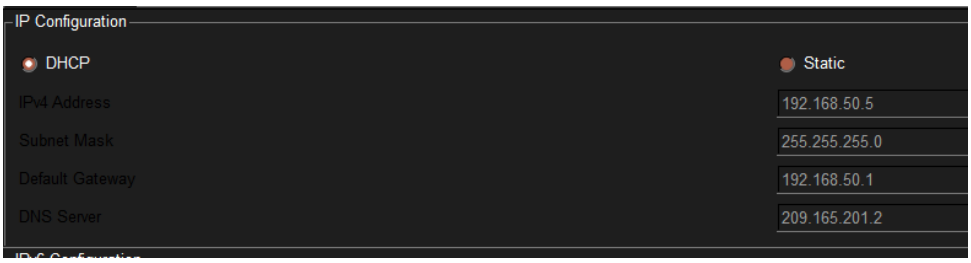
On PC-B (192.168.50.10)**:**

```
ping 192.168.10.10
```



Configure ASA DHCP Server

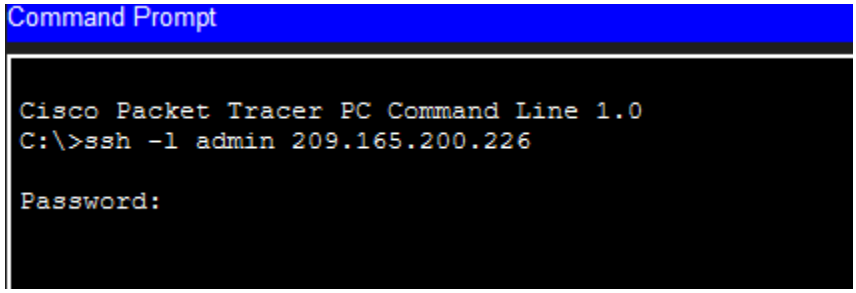
d. Test on PC-B



```
username admin password miu_admin privilege 15
aaa authentication ssh console LOCAL
crypto key generate rsa modulus 2048
ssh 192.168.50.0 255.255.255.0 INSIDE
ssh 192.168.10.10 255.255.255.255 OUTSIDE

ssh timeout 10
```

c. SSH test from PC1 → ASA OUTSIDE

From PC1:

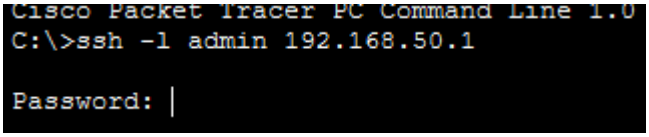


---

d. SSH test from PC-B → ASA INSIDE

From PC-B:

```
ssh -l admin 192.168.50.1
```



Configure the DMZ Interface (G1/3)

Commands:

```
configure terminal
interface gigabitEthernet1/3
 nameif DMZ
 security-level 60
 ip address 192.168.40.1 255.255.255.0
 no shutdown
exit
```

Verify:

```
show interface ip brief
show ip address
```

```
MIU-ASA#show ip address
System IP Addresses:
Interface              Name                IP address      Subnet mask     Method
GigabitEthernet1/1     OUTSIDE             209.165.200.226 255.255.255.248 manual
GigabitEthernet1/2     INSIDE              192.168.50.1    255.255.255.0   manual
GigabitEthernet1/3     DMZ                 192.168.40.1    255.255.255.0   manual
GigabitEthernet1/4                         unassigned      unassigned      unset
GigabitEthernet1/5                         unassigned      unassigned      unset
GigabitEthernet1/6                         unassigned      unassigned      unset
GigabitEthernet1/7                         unassigned      unassigned      unset
GigabitEthernet1/8                         unassigned      unassigned      unset
Management1/1                              unassigned      unassigned      unset

Current IP Addresses:
Interface              Name                IP address      Subnet mask     Method
GigabitEthernet1/1     OUTSIDE             209.165.200.226 255.255.255.248 manual
GigabitEthernet1/2     INSIDE              192.168.50.1    255.255.255.0   manual
GigabitEthernet1/3     DMZ                 192.168.40.1    255.255.255.0   manual
```

- **G1/3 up/up**
- **Nameif DMZ**
- **IP 192.168.40.1/24**
- **Security-level 60**

Step 3: Configure Static NAT for DMZ Server

DMZ Server = 192.168.40.10
Public IP to use = 209.165.200.227

Commands:

```
object network DMZ-SERVER
 host 192.168.40.10
 nat (DMZ,OUTSIDE) static 209.165.200.227
exit
```

This creates:

```
MIU-ASA#show nat
Auto NAT Policies (Section 2)
1 (DMZ) to (OUTSIDE) source static DMZ-SERVER 209.165.200.227
    translate_hits = 0, untranslate_hits = 0
2 (INSIDE) to (OUTSIDE) source dynamic INSIDE-NET interface
    translate_hits = 0, untranslate_hits = 0

MIU-ASA#
```

- Real IP**: 192.168.40.10**
- Mapped (public) IP**: 209.165.200.227**
- **Direction: DMZ → OUTSIDE**

Step 4: ACL to Allow Internet Access to the DMZ Server

- **Allow** TCP port 80 **(HTTP)**
- **Allow** ICMP
- **ACL name:** OUTSIDE-DMZ
- **Must permit traffic to the** private address **(ASA-specific rule)**

Commands:

```
access-list OUTSIDE-DMZ extended permit tcp any host 192.168.40.10 eq 80
access-list OUTSIDE-DMZ extended permit icmp any host 192.168.40.10
access-group OUTSIDE-DMZ in interface OUTSIDE
```

This applies the ACL inbound **on the OUTSIDE interface.**

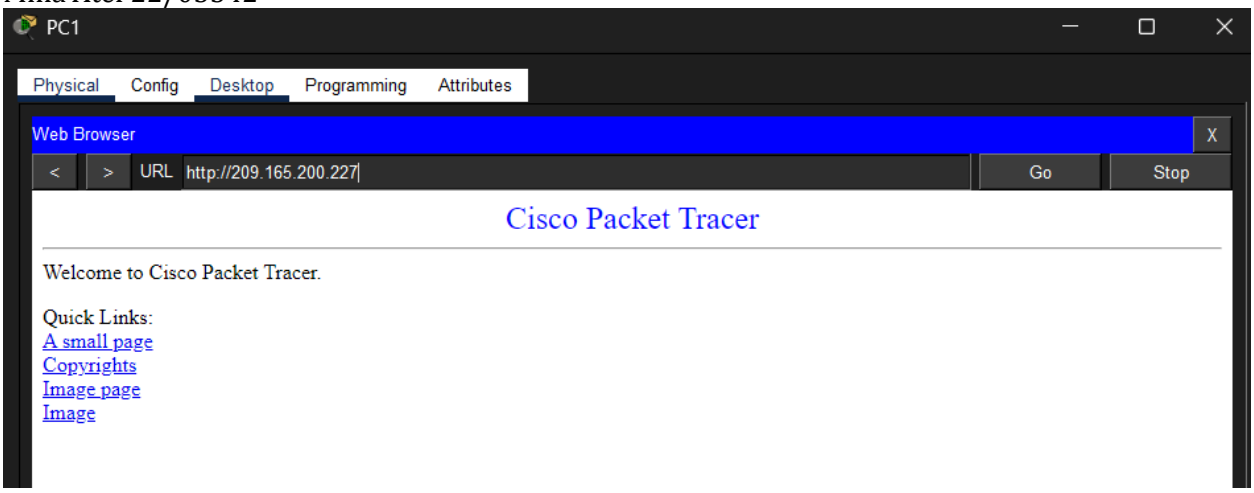
1. **Open browser:**

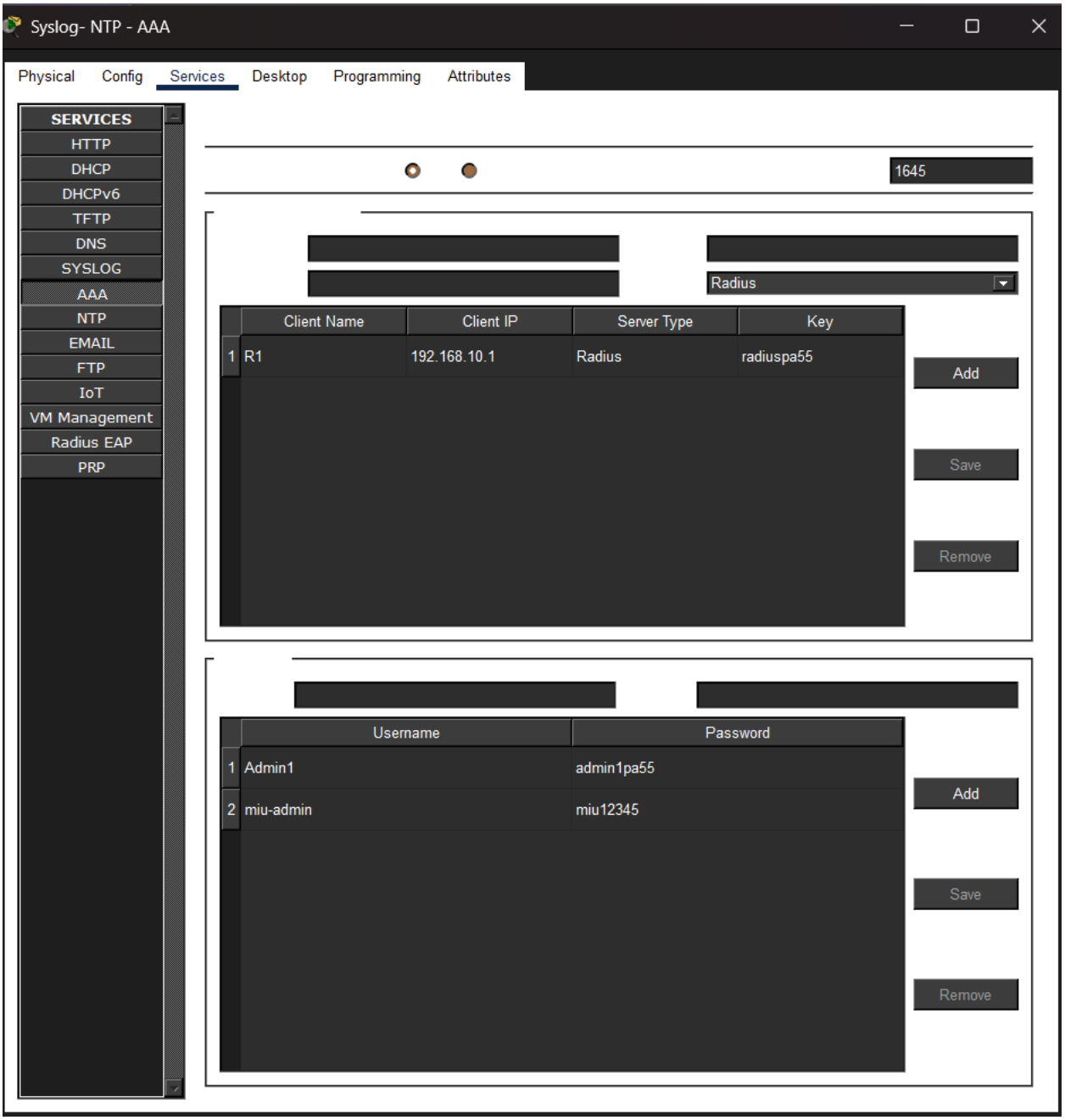
   ```
   http://209.165.200.227
   ```

2. **Ping the public IP:**

   ```
   ping 209.165.200.227
   ```

```
MIU-ASA#show run inter
MIU-ASA#show run interface g1/3
interface GigabitEthernet1/3
 nameif DMZ
 security-level 60
 ip address 192.168.40.1 255.255.255.0
!
MIU-ASA#show nat
Auto NAT Policies (Section 2)
1 (DMZ) to (OUTSIDE) source static DMZ-SERVER 209.165.200.227
    translate_hits = 0, untranslate_hits = 0
2 (INSIDE) to (OUTSIDE) source dynamic INSIDE-NET interface
    translate_hits = 4, untranslate_hits = 0

MIU-ASA#show xl
MIU-ASA#show xlate
1 in use, 1 most used
Flags: D - DNS, e - extended, I - identity, i - dynamic, r - portmap, s - static, T - twice, N - net-to-net
NAT from DMZ:192.168.40.10/32 to OUTSIDE:209.165.200.227/32 flags s idle 00:05:56,  timeout 0:00:00
```

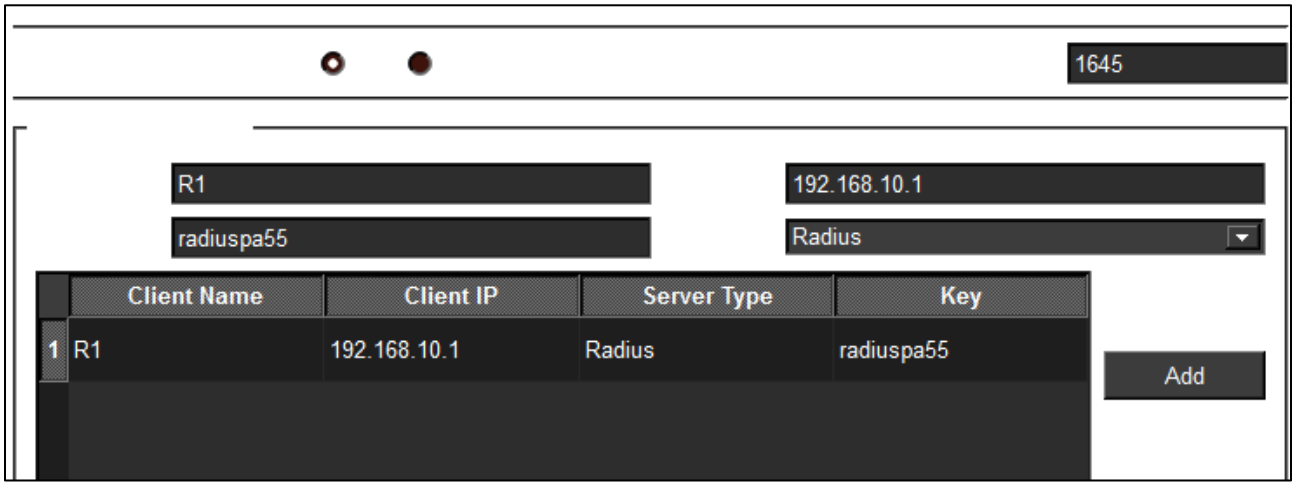

PC1
Physical
Config
Desktop
Programming
Attributes
Web Browser
X
<
>
URL
http://209.165.200.227
Go
Stop
Cisco Packet Tracer
Welcome to Cisco Packet Tracer.
Quick Links:
A small page
Copyrights
Image page
Image


Syslog- NTP - AAA
Physical
Config
Services
Desktop
Programming
Attributes
SERVICES
HTTP
DHCP
DHCPv6
TFTP
DNS
SYSLOG
AAA
NTP
EMAIL
FTP
IoT
VM Management
Radius EAP
PRP
1645
Radius
Client Name
Client IP
Server Type
Key
1
R1
192.168.10.1
Radius
radiuspa55
Add
Save
Remove
Username
Password
1
Admin1
admin1pa55
2
miu-admin
miu12345
Add
Save
Remove

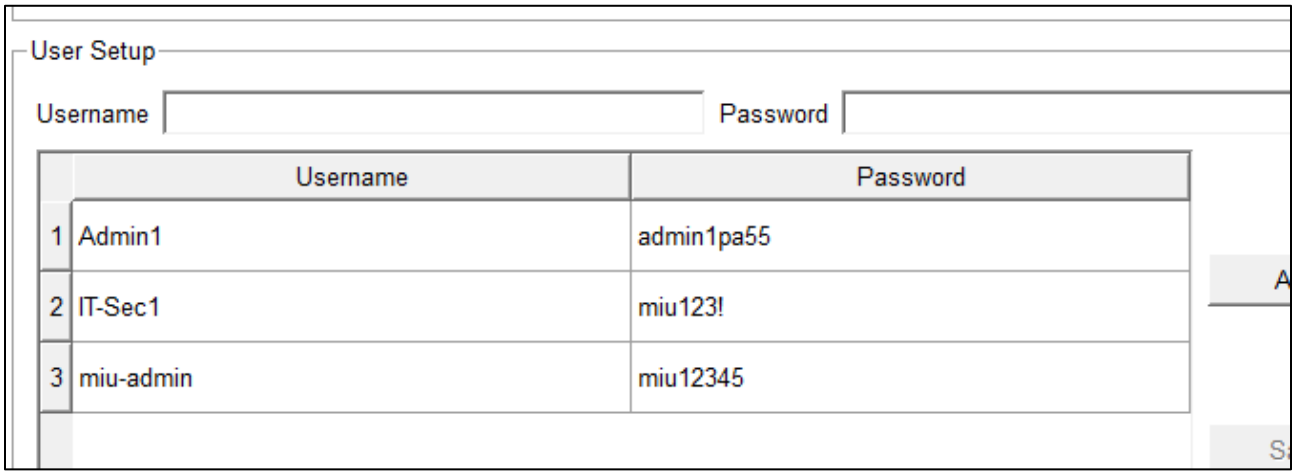
PART 10 – RADIUS AAA ON R1

1 Configure the RADIUS Server (GUI)

1. Click the **RADIUS Server**.
2. Go to **Services → AAA**.
3. Make sure:
   - **AAA service = ON**
   - Under **Network Configuration**:
     - Device name: **R1**
     - Key: **radiuspa55**



   - Under **User Setup**:
     - Username: **IT-Sec1**
     - Password: **miu123!**



✓ close the server.

2 Go to Router R1 (CLI)

```
enable
conf t
username Admin1 secret admin1pa55
ip domain-name miu.edu.eg
crypto key generate rsa modulus 1024
radius-server host 192.168.10.20 key radiuspa55
aaa new-model
aaa authentication login default group radius local
line console 0
 login authentication default
 exit
line vty 0 4
 transport input ssh
 login authentication default
 exit
```

a) RADIUS login

- Username: **IT-Sec1**
- Password: **miu123!**

b) Local fallback login

- Username: **Admin1**
- Password: **admin1pa55**

```
User Access Verification

Username: Admin1
Password:
R1>
```

Summary:Part 10 sets up login security on R1 using a RADIUS server, with a local user as a backup if the server is unavailable.

Part 11 – Server Configuration

FTP Server:

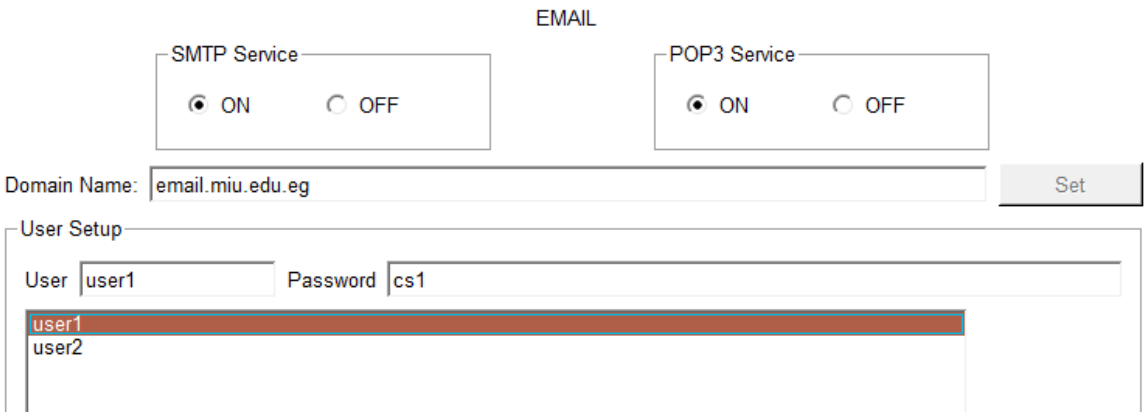
1. **Click the server**
2. **Go to** Services
3. **Select** FTP
4. **Turn** ON **the FTP service**

Under FTP user accounts:

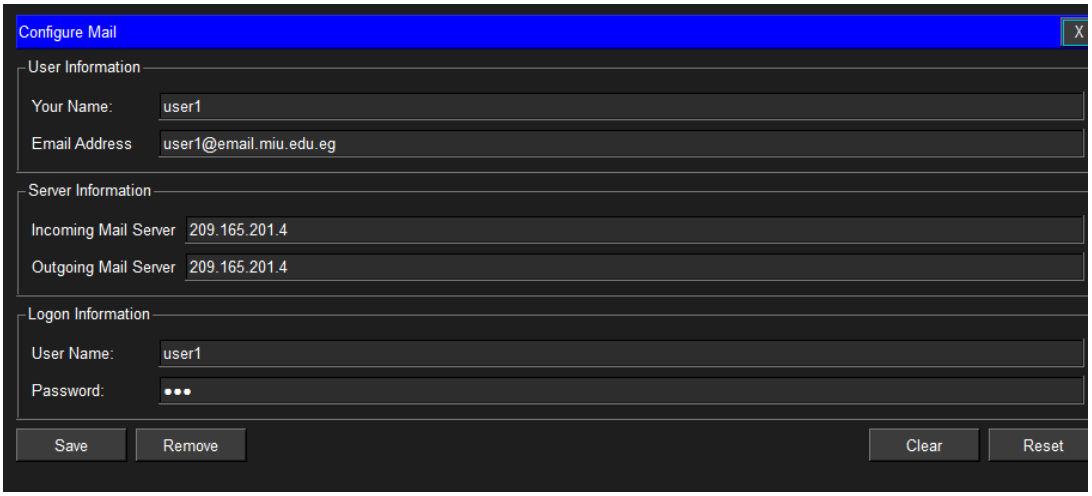
Username: CS_25
Password: miu123

```
C:\>ftp 209.165.201.4
Trying to connect...209.165.201.4
Connected to 209.165.201.4
220- Welcome to PT Ftp server
Username:CS_25
331- Username ok, need password
Password:******
230- Logged in
(passive mode On)
ftp>


ftp>put test.txt

Writing file test.txt to 209.165.201.4:
File transfer in progress...

[Transfer complete - 9 bytes]

9 bytes copied in 0.021 secs (428 bytes/sec)
```

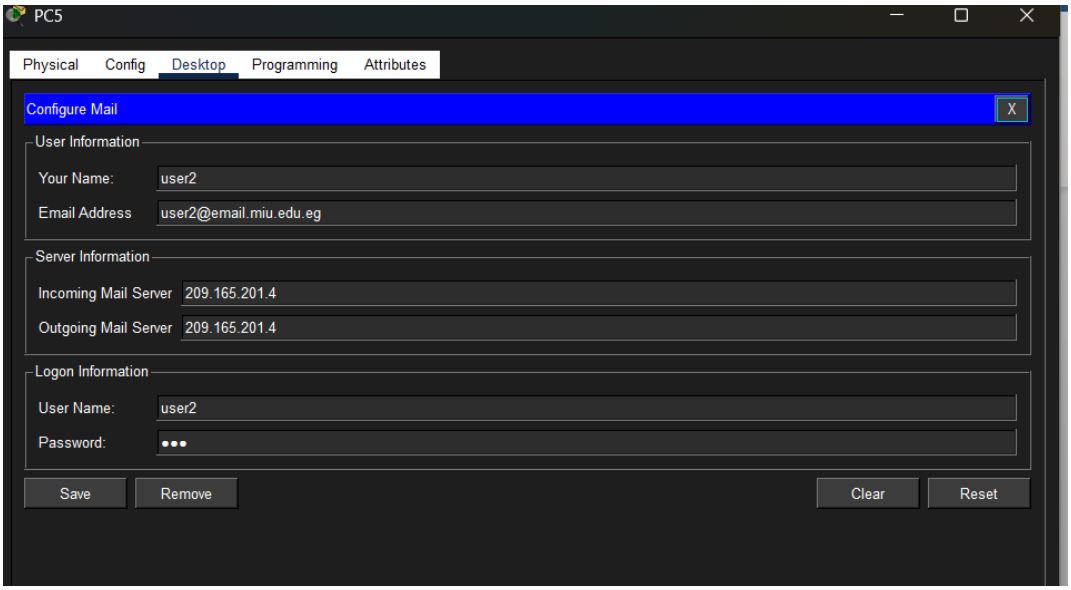
Email Server Configuration



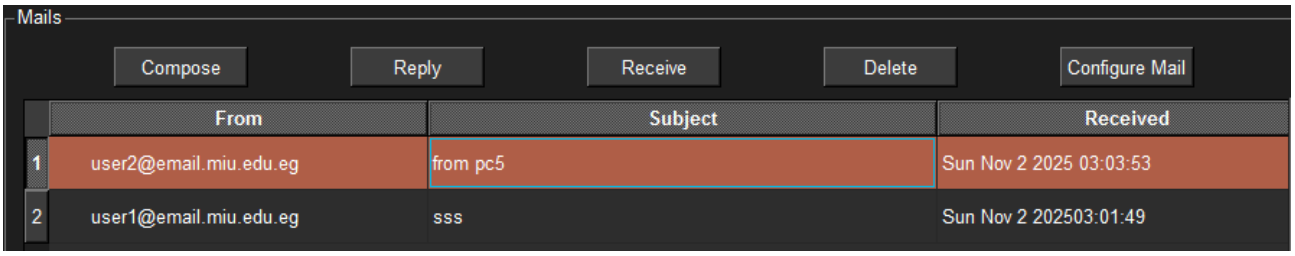
From PC1**:**



From PC5**:**

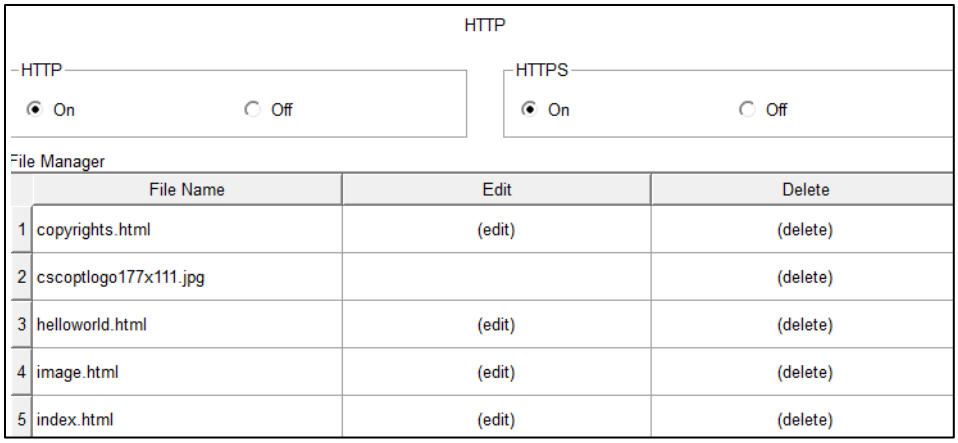


s



Both send and receive messages successfully.

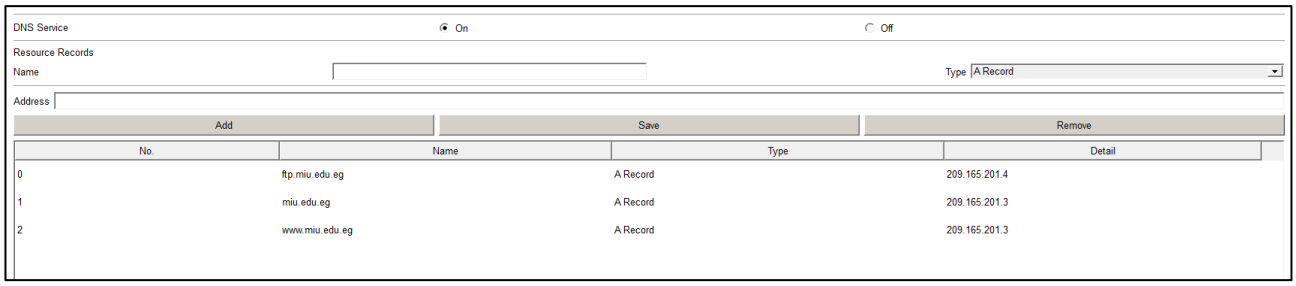
Web Server Configuration

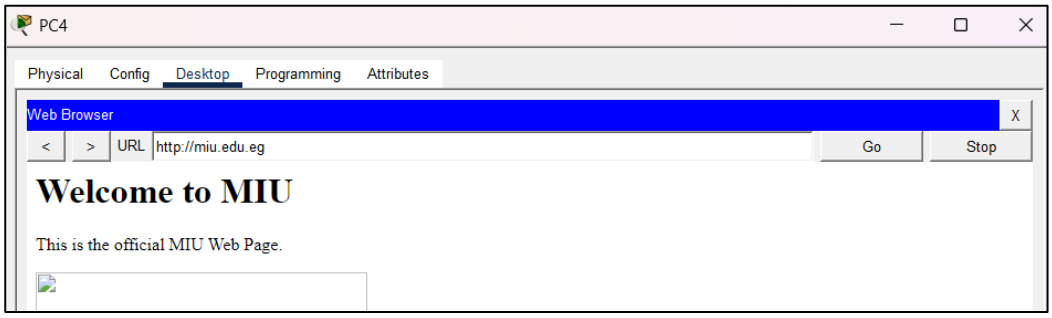


	File Name	Edit	Delete
1	copyrights.html	(edit)	(delete)
2	cscoptlogo177x111.jpg		(delete)
3	helloworld.html	(edit)	(delete)
4	image.html	(edit)	(delete)
5	index.html	(edit)	(delete)

verify that is working

DNS Server Configuration





Part 12.1 — SDN Controller Network Management

Step 1 — Configure PT-Controller0

a. open the controller

PT-Controller0 → Config

b. set the gateway

in Gateway/DNS IPv4, choose Static
set:

- Default Gateway: `192.168.10.1`
- DNS Server: `209.165.201.2`

c. go to the correct interface

INTERFACE → GigabitEthernet0

d. assign the IP

set:

- IP Address: `192.168.10.30`
- Subnet Mask: `255.255.255.0`

make sure Port Status = ON

e. enable the controller engine

REAL WORLD → Controller

if the field says Disabled in Preferences, then:

1. Options → Preferences
2. Miscellaneous
3. enable: External Network Access for Network Controller REST API
4. close Preferences
5. return to PT-Controller0 → Config → Controller

f. start the controller

click Access Enabled
then confirm:

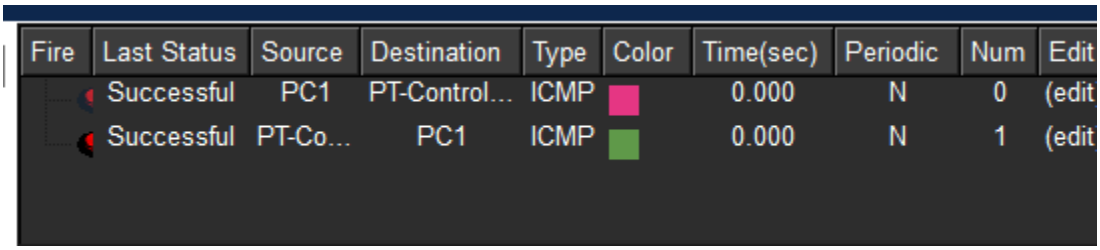
- Server Status: Listening on port 58000

if it shows a different port, change it manually to 58000.

Step 2 — Test from PC1

on PC1 → Desktop → Command Prompt:

```
ping 192.168.10.30
```



Step 3 — Register SDN user

1. PC1 → Desktop → Web Browser
2. enter:

```
http://192.168.10.30
```

3. fill the setup page:

- Username: IT-Sec1
- Password: miu123!

click SETUP

4. login with:

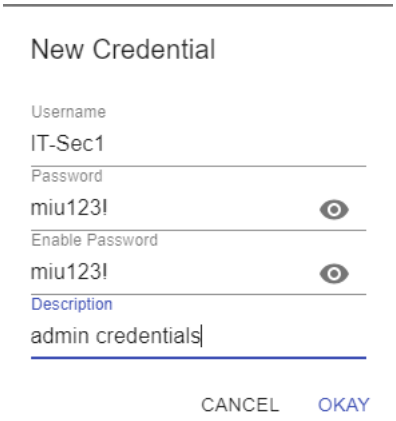
- Username: IT-Sec1
- Password: miu123!

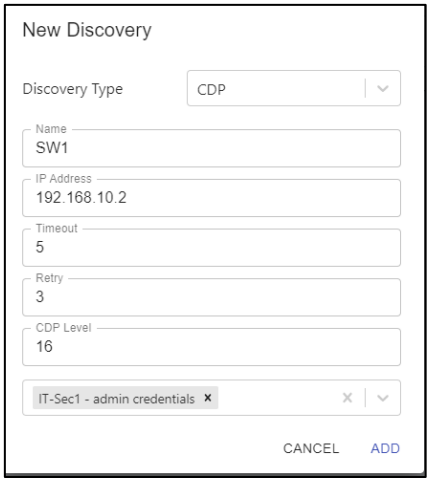
5. should now be inside the PT-Controller dashboard

Mina Atef 22/05542

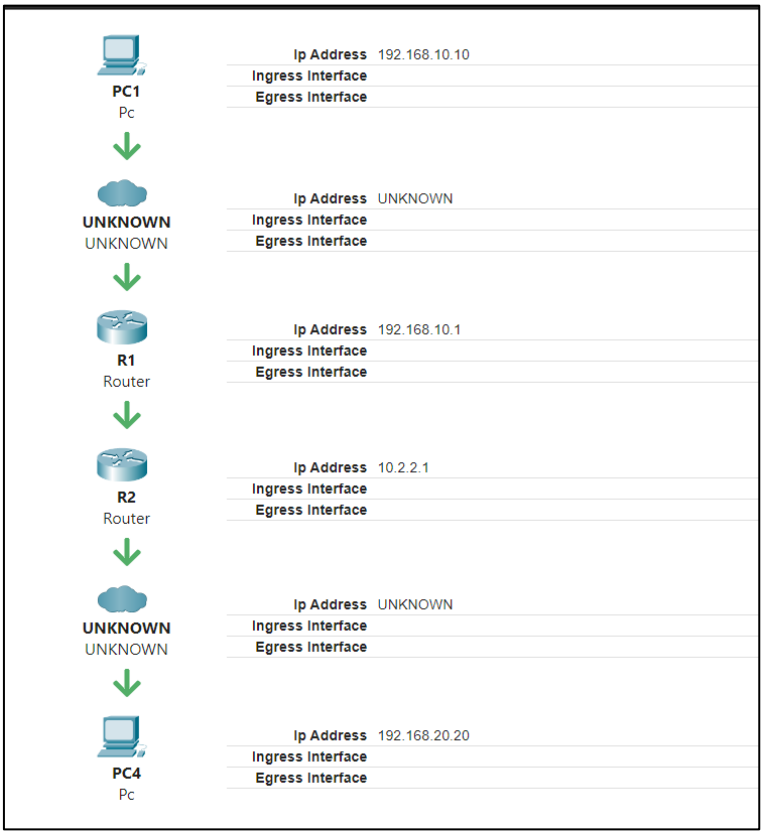
Use SDN Controller to Discover Topology

Start to discovery

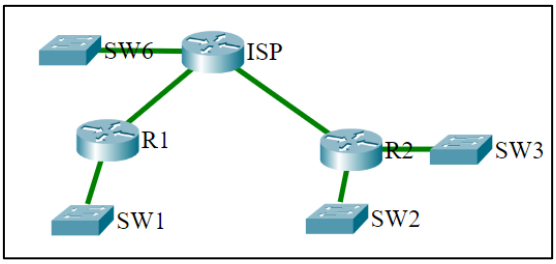




This is the path it take when it ping from pc1 to pc4 and the unnkown is the switch this is normal screen shot is cropped



Our Topology

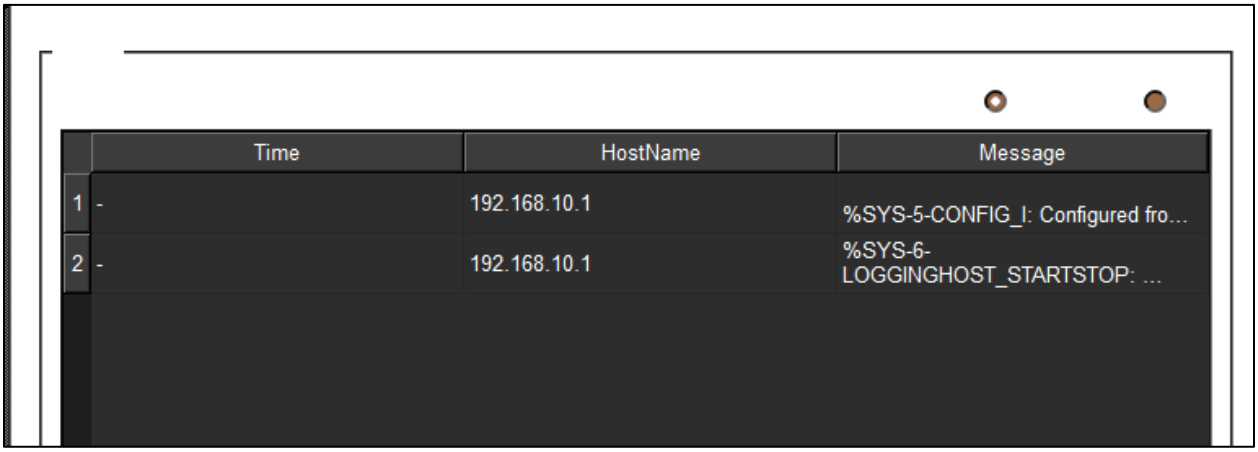


note the ASA and the switch below it is don't support the pt controller so therefore it doesn't appear in the Topology

Mina Atef 22/05542
This is the configuration of the PT-Controller

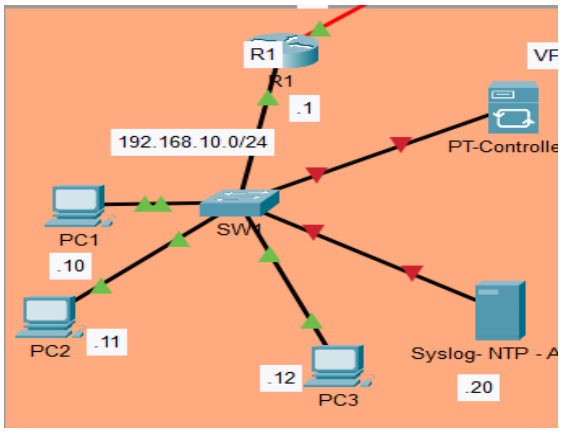
This is shows that syslog is working fine

SYSLOG



Troubleshooting

a)The PT-Controller0 and SYS-NTP -AAA are red connection to SW1



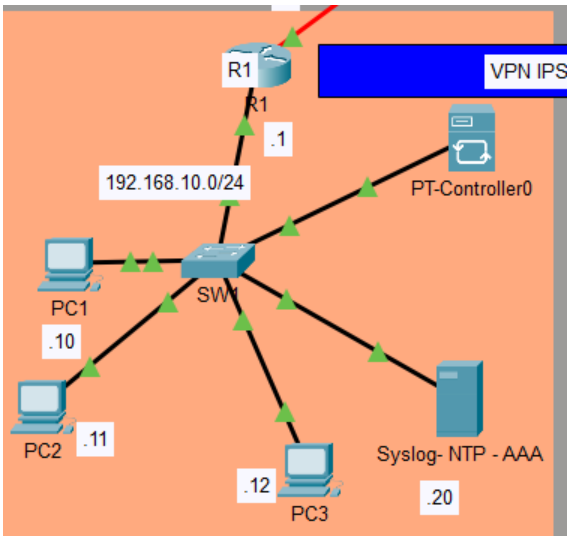
1-i checked IP/Mask/GW all correct

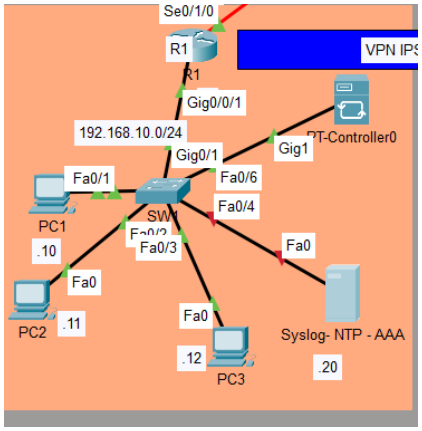
```
SW1(config-if)#no shutdown

SW1(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/6, changed state to up
```

2-enable interface show in the option perferneces

3-interface f0/4 ,f0/6 changed it no shutdown

```
SW1(config)#interface f0/4
SW1(config-if)#no sh
SW1(config-if)#no shutdown

SW1(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/4, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed state to up
```





B)the ssh ISP not working
try to configure OSPF again

```
C:\>ssh -l IT-Sec1 10.1.1.2

% Connection timed out; remote host not responding
C:\>
```

```
R1#show crypto ipsec sa

interface: Serial0/1/0
    Crypto map tag: VPN-MAP, local addr 10.1.1.1

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (192.168.10.0/255.255.255.0/0/0)
   remote  ident (addr/mask/prot/port): (192.168.20.0/255.255.255.0/0/0)
   current_peer 10.2.2.1 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 0, #pkts encrypt: 0, #pkts digest: 0
   #pkts decaps: 0, #pkts decrypt: 0, #pkts verify: 0
   #pkts compressed: 0, #pkts decompressed: 0
   #pkts not compressed: 0, #pkts compr. failed: 0
   #pkts not decompressed: 0, #pkts decompress failed: 0
   #send errors 0, #recv errors 0

     local crypto endpt.: 10.1.1.1, remote crypto endpt.:10.2.2.1
     path mtu 1500, ip mtu 1500, ip mtu idb Serial0/1/0
     current outbound spi: 0x0(0)

     inbound esp sas:
```

key chain name is MTU-SEC but it should be MIU_SEC

After I changed it it worked fine