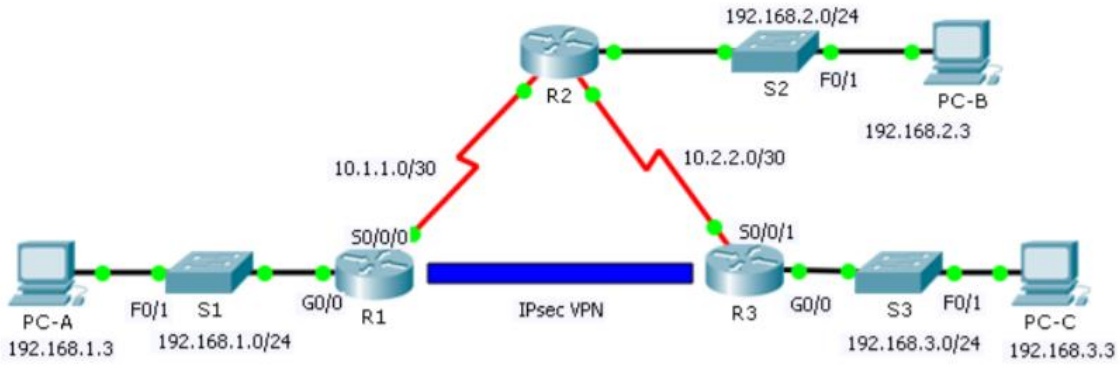
Aryaman Mishra

19BCE1027

VPN Configuration



Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/0	192.168.1.1	255.255.255.0	N/A
	S0/0/0	10.1.1.2	255.255.255.252	N/A
R2	G0/0	192.168.2.1	255.255.255.0	N/A
	S0/0/0	10.1.1.1	255.255.255.252	N/A
	S0/0/1	10.2.2.1	255.255.255.252	N/A
R3	G0/0	192.168.3.1	255.255.255.0	N/A
	S0/0/1	10.2.2.2	255.255.255.252	N/A
PC-A	NIC	192.168.1.3	255.255.255.0	192.168.1.1
PC-B	NIC	192.168.2.3	255.255.255.0	192.168.2.1
PC-C	NIC	192.168.3.3	255.255.255.0	192.168.3.1

1. Starting configurations for R1, ISP, and R3. Paste to global config mode :

hostname R1

interface g0/1

ip address 192.168.1.1 255.255.255.0

no shut

interface g0/0

ip address 209.165.100.1 255.255.255.0

no shut

exit

ip route 0.0.0.0 0.0.0.0 209.165.100.2

hostname ISP

interface g0/1

ip address 209.165.200.2 255.255.255.0

no shut

interface g0/0

ip address 209.165.100.2 255.255.255.0

no shut

exit

hostname R3

interface g0/1

ip address 192.168.3.1 255.255.255.0

no shut

interface g0/0

ip address 209.165.200.1 255.255.255.0

no shut

exit

ip route 0.0.0.0 0.0.0.0 209.165.200.2

2. Make sure routers have the security license enabled:

show version

license boot module c1900 technology-package securityk9

copy run start

reload

3. Configure IPsec on the routers at each end of the tunnel (R1 and R3)

!R1

crypto isakmp policy 10

```
encryption aes 256
authentication pre-share
group 5
!
crypto isakmp key secretkey address 209.165.200.1
!
crypto ipsec transform-set R1-R3 esp-aes 256 esp-sha-hmac
!
crypto map IPSEC-MAP 10 ipsec-isakmp
set peer 209.165.200.1
set pfs group5
set security-association lifetime seconds 86400
set transform-set R1-R3
match address 100
!
interface GigabitEthernet0/0
crypto map IPSEC-MAP
!
access-list 100 permit ip 192.168.1.0 0.0.0.255 192.168.3.0 0.0.0.255

!R3
crypto isakmp policy 10
encryption aes 256
authentication pre-share
group 5
!
crypto isakmp key secretkey address 209.165.100.1
!
crypto ipsec transform-set R3-R1 esp-aes 256 esp-sha-hmac
!
crypto map IPSEC-MAP 10 ipsec-isakmp
```

set peer 209.165.100.1

set pfs group5

set security-association lifetime seconds 86400

set transform-set R3-R1

match address 100

!

interface GigabitEthernet0/0

crypto map IPSEC-MAP

!

R3

access-list 100 permit ip 192.168.3.0 0.0.0.255 192.168.1.0 0.0.0.255

R1

access-list 100 permit ip 192.168.1.0 0.0.0.255 192.168.3.0 0.0.0.255

Step 1: Test connectivity.

Ping from **PC-A** to **PC-C**.

Step 2: Identify interesting traffic on R1.

Configure ACL 110 to identify the traffic from the LAN on **R1** to the LAN on **R3** as interesting. This interesting traffic will trigger the IPsec VPN to be implemented whenever there is traffic between **R1** to **R3** LANs. All other traffic sourced from the LANs will not be encrypted. Remember that due to the implicit deny any, there is no need to add the statement to the list.

```
R1(config)# access-list 110 permit ip 192.168.1.0 0.0.0.255 192.168.3.0
0.0.0.255
```

Step 3: Configure the ISAKMP Phase 1 properties on R1.

Configure the crypto ISAKMP policy **10** properties on **R1** along with the shared crypto key **cisco**. Refer to the ISAKMP Phase 1 table for the specific parameters to configure. Default values do not have to be configured therefore only the encryption, key exchange method, and DH method must be configured.

```
R1(config)# crypto isakmp policy 10
R1(config-isakmp)# encryption aes
R1(config-isakmp)# authentication pre-share
R1(config-isakmp)# group 2
R1(config-isakmp)# exit
R1(config)# crypto isakmp key cisco address 10.2.2.2
```

Step 4: Configure the ISAKMP Phase 2 properties on R1.

Create the transform-set **VPN-SET** to use **esp-3des** and **esp-sha-hmac**. Then create the crypto map **VPN-MAP** that binds all of the Phase 2 parameters together. Use sequence number **10** and identify it as an **ipsec-isakmp** map.

```
R1(config)# crypto ipsec transform-set VPN-SET esp-3des esp-sha-hmac
R1(config)# crypto map VPN-MAP 10 ipsec-isakmp
R1(config-crypto-map)# description VPN connection to R3
R1(config-crypto-map)# set peer 10.2.2.2
R1(config-crypto-map)# set transform-set VPN-SET
R1(config-crypto-map)# match address 110
R1(config-crypto-map)# exit
```

Step 5: Configure the crypto map on the outgoing interface.

Finally, bind the **VPN-MAP** crypto map to the outgoing Serial 0/0/0 interface. **Note:** This is not graded.

```
R1(config)# interface S0/0/0
R1(config-if)# crypto map VPN-MAP
```

Part 3: Configure IPsec Parameters on R3

Step 1: Configure router R3 to support a site-to-site VPN with R1.

Now configure reciprocating parameters on **R3**. Configure ACL **110** identifying the traffic from the LAN on **R3** to the LAN on **R1** as interesting.

```
R3(config)# access-list 110 permit ip 192.168.3.0 0.0.0.255 192.168.1.0
0.0.0.255
```

Step 2: Configure the ISAKMP Phase 1 properties on R3.

Configure the crypto ISAKMP policy **10** properties on **R3** along with the shared crypto key **cisco**.

```
R3(config)# crypto isakmp policy 10
R3(config-isakmp)# encryption aes
R3(config-isakmp)# authentication pre-share
R3(config-isakmp)# group 2
R3(config-isakmp)# exit
R3(config)# crypto isakmp key cisco address 10.1.1.2
```

Step 3: Configure the ISAKMP Phase 2 properties on R1.

Like you did on **R1**, create the transform-set **VPN-SET** to use **esp-3des** and **esp-sha-hmac**. Then create the crypto map **VPN-MAP** that binds all of the Phase 2 parameters together. Use sequence number **10** and identify it as an **ipsec-isakmp** map.

```
R3(config)# crypto ipsec transform-set VPN-SET esp-3des esp-sha-hmac
R3(config)# crypto map VPN-MAP 10 ipsec-isakmp
R3(config-crypto-map)# description VPN connection to R1
R3(config-crypto-map)# set peer 10.1.1.2
R3(config-crypto-map)# set transform-set VPN-SET
R3(config-crypto-map)# match address 110
R3(config-crypto-map)# exit
```

Step 4: Configure the crypto map on the outgoing interface.

Finally, bind the **VPN-MAP** crypto map to the outgoing Serial 0/0/1 interface. **Note**: This is not graded.

```
R3(config)# interface S0/0/1
R3(config-if)# crypto map VPN-MAP
```

Part 4: Verify the IPsec VPN

Step 1: Verify the tunnel prior to interesting traffic.

Issue the **show crypto ipsec sa** command on **R1**. Notice that the number of packets encapsulated, encrypted, decapsulated and decrypted are all set to 0.

```
R1# show crypto ipsec sa

interface: Serial0/0/0
    Crypto map tag: VPN-MAP, local addr 10.1.1.2

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
   remote  ident (addr/mask/prot/port): (192.168.3.0/255.255.255.0/0/0)
   current_peer 10.2.2.2 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.: 10.1.1.2, remote crypto endpt.:10.2.2.2
     path mtu 1500, ip mtu 1500, ip mtu idb Serial0/0/0
     current outbound spi: 0x0(0)
<output omitted>
```

Step 2: Create interesting traffic.

Ping **PC-C** from **PC-A**.

Step 3: Verify the tunnel after interesting traffic.

On **R1**, re-issue the **show crypto ipsec sa** command. Now notice that the number of packets is more than 0 indicating that the IPsec VPN tunnel is working.

```
R1# show crypto ipsec sa

interface: Serial0/0/0
    Crypto map tag: VPN-MAP, local addr 10.1.1.2

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
   remote  ident (addr/mask/prot/port): (192.168.3.0/255.255.255.0/0/0)
   current_peer 10.2.2.2 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 3, #pkts encrypt: 3, #pkts digest: 0
   #pkts decaps: 3, #pkts decrypt: 3, #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.: 10.1.1.2, remote crypto endpt.:10.2.2.2
     path mtu 1500, ip mtu 1500, ip mtu idb Serial0/0/0
     current outbound spi: 0x0A496941(172583233)
<output omitted>
```

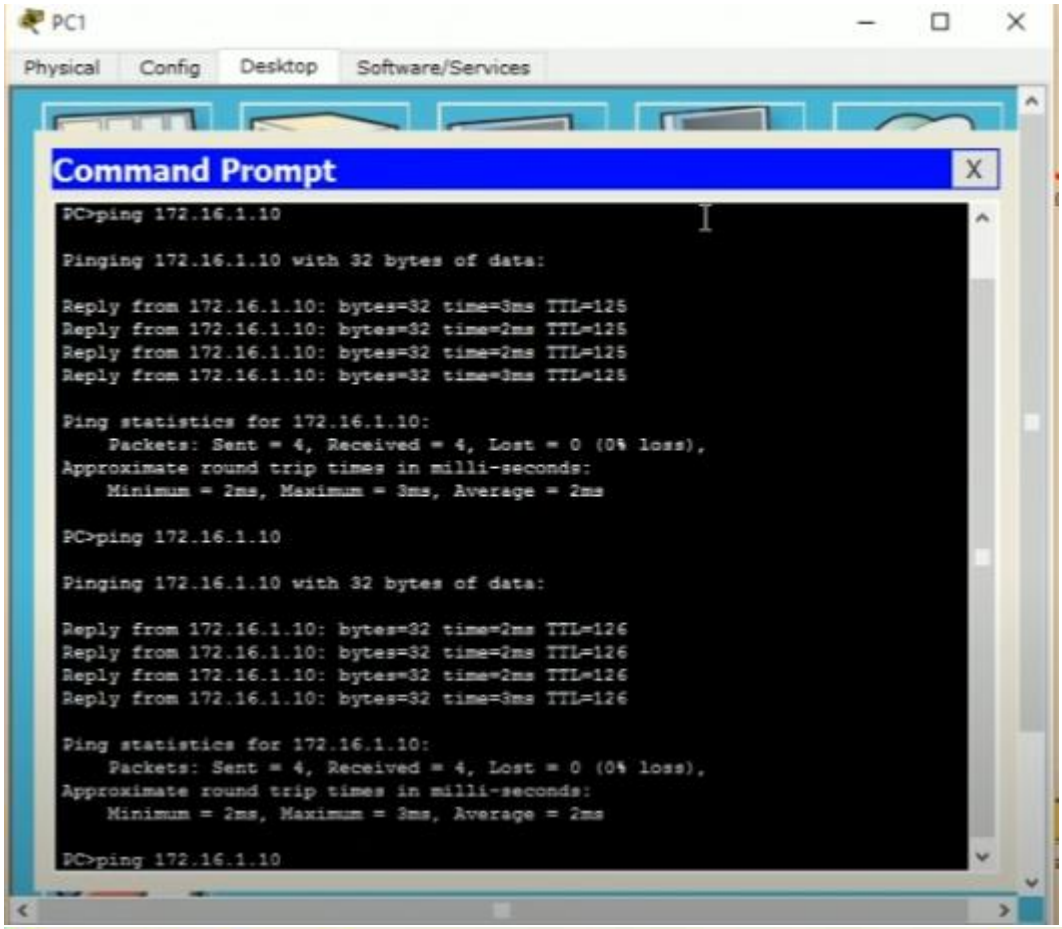
Step 4: Create uninteresting traffic.

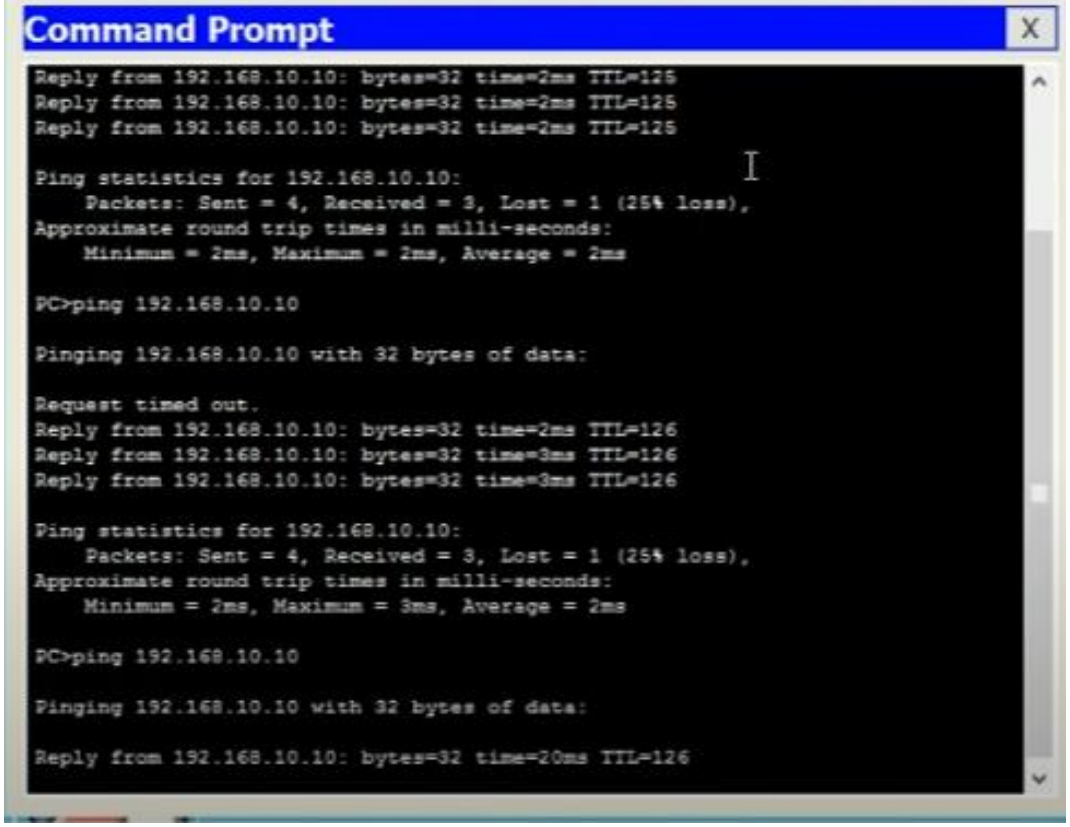
Ping **PC-B** from **PC-A**.

Step 5: Verify the tunnel.

On **R1**, re-issue the **show crypto ipsec sa** command. Finally, notice that the number of packets has not changed verifying that uninteresting traffic is not encrypted.

Output on next Page:


PC1
Physical
Config
Desktop
Software/Services
Command Prompt
PC>ping 172.16.1.10
Pinging 172.16.1.10 with 32 bytes of data:
Reply from 172.16.1.10: bytes=32 time=3ms TTL=125
Reply from 172.16.1.10: bytes=32 time=2ms TTL=125
Reply from 172.16.1.10: bytes=32 time=2ms TTL=125
Reply from 172.16.1.10: bytes=32 time=3ms TTL=125
Ping statistics for 172.16.1.10:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 3ms, Average = 2ms
PC>ping 172.16.1.10
Pinging 172.16.1.10 with 32 bytes of data:
Reply from 172.16.1.10: bytes=32 time=2ms TTL=126
Reply from 172.16.1.10: bytes=32 time=2ms TTL=126
Reply from 172.16.1.10: bytes=32 time=2ms TTL=126
Reply from 172.16.1.10: bytes=32 time=3ms TTL=126
Ping statistics for 172.16.1.10:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 3ms, Average = 2ms
PC>ping 172.16.1.10


Command Prompt
Reply from 192.168.10.10: bytes=32 time=2ms TTL=125
Reply from 192.168.10.10: bytes=32 time=2ms TTL=125
Reply from 192.168.10.10: bytes=32 time=2ms TTL=125
Ping statistics for 192.168.10.10:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 2ms, Average = 2ms
PC>ping 192.168.10.10
Pinging 192.168.10.10 with 32 bytes of data:
Request timed out.
Reply from 192.168.10.10: bytes=32 time=2ms TTL=126
Reply from 192.168.10.10: bytes=32 time=3ms TTL=126
Reply from 192.168.10.10: bytes=32 time=3ms TTL=126
Ping statistics for 192.168.10.10:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 3ms, Average = 2ms
PC>ping 192.168.10.10
Pinging 192.168.10.10 with 32 bytes of data:
Reply from 192.168.10.10: bytes=32 time=20ms TTL=126