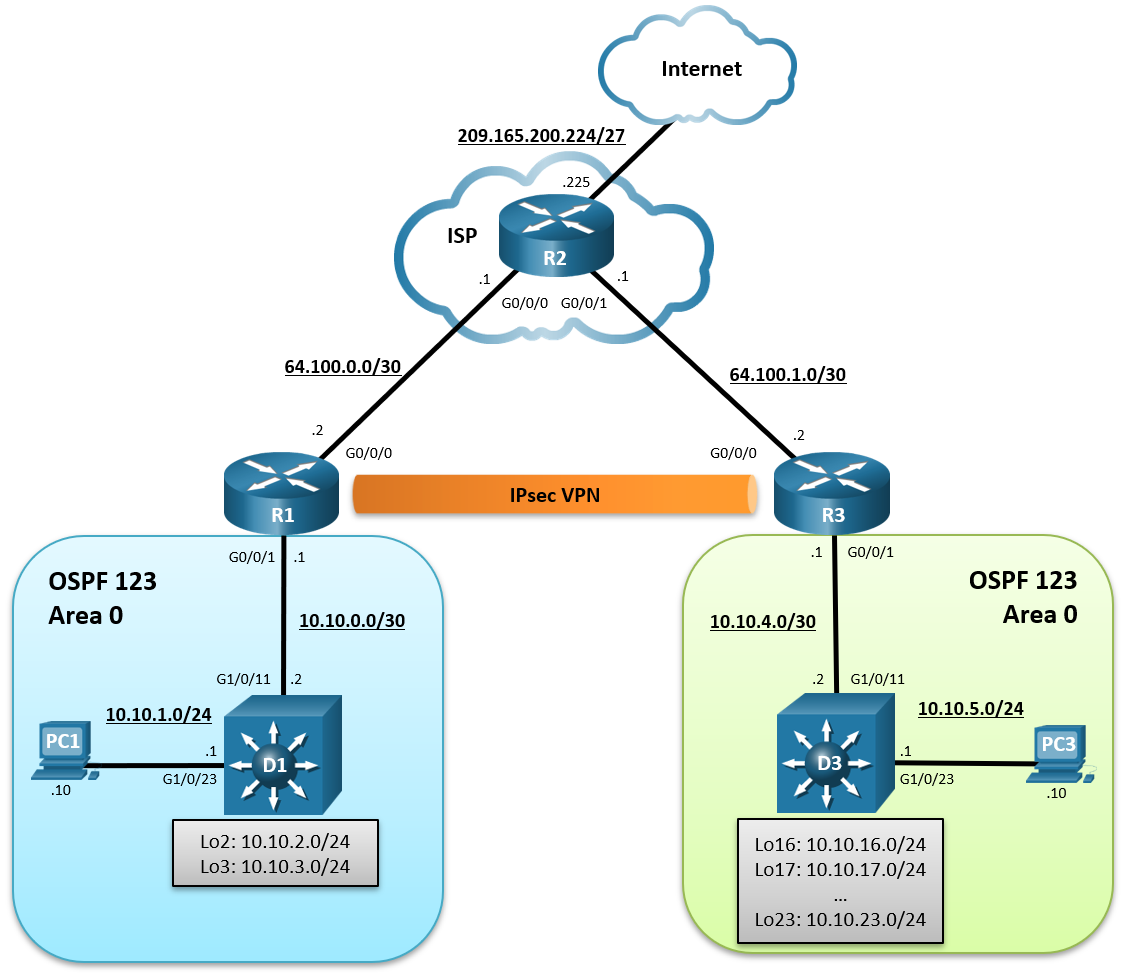
Lab - Implement IPsec Site-to-Site VPNs (Instructor Version)

**Instructor Note**: Red font color or gray highlights indicate text that appears in the instructor copy only.

# Topology



# Addressing Table

|  |  |  |  |
| --- | --- | --- | --- |
| Device | Interface | IPv4 Address | Default Gateway |
| R1 | G0/0/0 | 64.100.0.2/30 | N/A |
| R1 | G0/0/1 | 10.10.0.1/29 | N/A |
| R2 | G0/0/0 | 64.100.0.1/30 | N/A |
| R2 | G0/0/1 | 64.100.1.1/30 | N/A |
| R2 | Lo0 | 209.165.200.225 | N/A |
| R3 | G0/0/0 | 64.100.1.2/30 | N/A |
| R3 | G0/0/1 | 10.10.4.1/30 | N/A |
| D1 | G1/0/11 | 10.10.0.2/29 | N/A |
| D1 | G1/0/23 | 10.10.1.1/24 | N/A |
| D1 | Lo2 | 10.10.2.1/24 | N/A |
| D1 | Lo3 | 10.10.3.1/24 | N/A |
| D3 | G1/0/11 | 10.10.0.3/29 | N/A |
| D3 | G1/0/23 | 10.10.5.1/24 | N/A |
| D3 | Lo16 | 10.10.16.1/24 | N/A |
| D3 | Lo17 | 10.10.17.1/24 | N/A |
| D3 | Lo18 | 10.10.18.1/24 | N/A |
| D3 | Lo19 | 10.10.19.1/24 | N/A |
| D3 | Lo20 | 10.10.20.1/24 | N/A |
| D3 | Lo21 | 10.10.21.1/24 | N/A |
| D3 | Lo22 | 10.10.22.1/24 | N/A |
| D3 | Lo23 | 10.10.23.1/24 | N/A |
| PC1 | NIC | 10.10.1.10/24 | 10.10.1.1 |
| PC3 | NIC | 10.10.5.10/24 | 10.10.5.1 |

# Objectives

Part 1: Build the Network, Configure Basic Device Settings and Static Routing

Part 2: Configure a Site-to-Site VPN using Crypto Maps Between R1 and R3

Part 3: Verify a Site-to-Site VPN Between R1 and R3

# Background / Scenario

VPNs provide a secure method of transmitting data over a public network, such as the internet. VPN connections help reduce the costs associated with leased lines. Site-to-site VPNs typically provide a secure (IPsec or other) tunnel between a branch office and a central office. Another common implementation of VPN technology is remote access to a corporate office from a telecommuter location, such as a small office or home office.

In this lab, you will establish a site-to-site IPsec VPN tunnel between R1 to R3 via R2. R2 is the ISP router, and it will have no knowledge of the VPN. IPsec provides secure transmission of sensitive information over unprotected networks, such as the internet. IPsec works at the network layer and protects and authenticates IP packets between participating IPsec devices (peers), such as Cisco routers.

**Note**: This lab is an exercise in developing, deploying, and verifying how VPNs operate and does not reflect networking best practices.

**Note**: The routers used with this CCNP hands-on lab are three Cisco 4221 and the two Layer 3 switches are Catalyst 3650 switches. Other routers and Layer 3 switches and Cisco IOS versions can be used. Depending on the model and Cisco IOS version, the commands available and the output produced might vary from what is shown in the labs.

**Note**: Ensure that the routers and switches have been erased and have no startup configurations. If you are unsure contact your instructor.

**Instructor Note**: Refer to the Instructor Lab Manual for the procedures to initialize and reload devices.

# Required Resources

* 3 Routers (Cisco 4221 with Cisco IOS XE Release 16.9.4 universal image or comparable)
* 2 Switches (Cisco 4221 with Cisco IOS XE Release 16.9.4 universal image or comparable)
* 2 PCs (Choice of operating system with a terminal emulation program installed)
* Console cables to configure the Cisco IOS devices via the console ports
* Ethernet cables as shown in the topology

# Instructions

## Build the Network, Configure Basic Device Settings and Static Routing

In Part 1, you will set up the network topology, configure basic settings, interface addressing, and single-area OSPFv2 on the routers.

### Cable the network as shown in the topology.

Attach the devices as shown in the topology diagram, and cable as necessary.

### Configure basic settings for the routers.

* + - 1. Console into each router and switch, enter global configuration mode, and apply the basic settings, and interface addressing. A command list for each device is provided for your reference.

Routing is enabled as follows:

* R2 has a static route to the networks connected to R1 (i.e., 10.10.0.0/22) and two static routes to the networks connected to R3 (i.e., 10.10.4.0/22, 10.10.16.0/21).
* R1 and R3 each have a default static route to R2.
* OSPFv2 routing is enabled between R1 and D1, and R1 is propagating the default route to D1.
* OSPFv2 routing is enabled between R3 and D3, and R3 is propagating the default route to D3.
* A command list for each router is listed below to perform initial configuration.

Router R1

Open configuration window

hostname R1

no ip domain lookup

line con 0

logging sync

exec-time 0 0

exit

banner motd # This is R1, Implement IPsec Site-to-Site VPNs #

interface g0/0/0

description Connection to R2

ip add 64.100.0.2 255.255.255.252

no shut

exit

interface GigabitEthernet0/0/1

description Connection to D1

ip address 10.10.0.1 255.255.255.252

no shut

exit

router ospf 123

router-id 1.1.1.1

auto-cost reference-bandwidth 1000

network 10.10.0.0 0.0.0.3 area 0

default-information originate

exit

ip route 0.0.0.0 0.0.0.0 64.100.0.1

Router R2

hostname R2

no ip domain lookup

line con 0

logging sync

exec-time 0 0

exit

banner motd # This is R2, Implement IPsec Site-to-Site VPNs #

interface g0/0/0

description Connection to R1

ip add 64.100.0.1 255.255.255.252

no shut

exit

interface GigabitEthernet0/0/1

description Connection to R3

ip address 64.100.1.1 255.255.255.252

no shut

exit

int lo0

description Internet simulated address

ip add 209.165.200.225 255.255.255.224

exit

ip route 0.0.0.0 0.0.0.0 Loopback0

ip route 10.10.0.0 255.255.252.0 64.100.0.2

ip route 10.10.4.0 255.255.252.0 64.100.1.2

ip route 10.10.16.0 255.255.248.0 64.100.1.2

Router R3

hostname R3

no ip domain lookup

line con 0

logging sync

exec-time 0 0

exit

banner motd # This is R3, Implement IPsec Site-to-Site VPNs #

interface g0/0/0

description Connection to R2

ip add 64.100.1.2 255.255.255.252

no shut

exit

interface GigabitEthernet0/0/1

description Connection to D3

ip address 10.10.4.1 255.255.255.252

no shut

exit

ip route 0.0.0.0 0.0.0.0 64.100.1.1

router ospf 123

router-id 3.3.3.1

auto-cost reference-bandwidth 1000

network 10.10.4.0 0.0.0.3 area 0

default-information originate

exit

Switch D1

hostname D1

no ip domain lookup

line con 0

exec-timeout 0 0

logging synchronous

exit

banner motd # This is D1, Implement IPsec Site-to-Site VPNs #

interface G1/0/11

description Connection to R1

no switchport

ip address 10.10.0.2 255.255.255.252

no shut

exit

interface G1/0/23

description Connection to PC1

no switchport

ip address 10.10.1.1 255.255.255.0

no shut

exit

int Lo2

description Loopback to simulate an OSPF network

ip add 10.10.2.1 255.255.255.0

ip ospf network point-to-point

exit

int Lo3

description Loopback to simulate an OSPF network

ip add 10.10.3.1 255.255.255.0

ip ospf network point-to-point

exit

ip routing

router ospf 123

router-id 1.1.1.2

auto-cost reference-bandwidth 1000

network 10.10.0.0 0.0.3.255 area 0

exit

int range G1/0/1 - 10, G1/0/12 - 22, G1/0/24

shut

exit

Switch D3

hostname D3

no ip domain lookup

line con 0

logging sync

exec-time 0 0

exit

banner motd # This is D3, Implement IPsec Site-to-Site VPNs #

interface G1/0/11

description Connection to R3

no switchport

ip address 10.10.4.2 255.255.255.252

no shut

exit

interface G1/0/23

description Connection to PC3

no switchport

ip address 10.10.5.1 255.255.255.0

no shut

exit

int Lo16

description Loopback to simulate an OSPF network

ip add 10.10.16.1 255.255.255.0

ip ospf network point-to-point

exit

int Lo17

description Loopback to simulate an OSPF network

ip add 10.10.17.1 255.255.255.0

ip ospf network point-to-point

exit

int Lo18

description Loopback to simulate an OSPF network

ip add 10.10.18.1 255.255.255.0

ip ospf network point-to-point

exit

int Lo19

description Loopback to simulate an OSPF network

ip add 10.10.19.1 255.255.255.0

ip ospf network point-to-point

exit

int Lo20

description Loopback to simulate an OSPF network

ip add 10.10.20.1 255.255.255.0

ip ospf network point-to-point

exit

int Lo21

description Loopback to simulate an OSPF network

ip add 10.10.21.1 255.255.255.0

ip ospf network point-to-point

exit

int Lo22

description Loopback to simulate an OSPF network

ip add 10.10.22.1 255.255.255.0

ip ospf network point-to-point

exit

int Lo23

description Loopback to simulate an OSPF network

ip add 10.10.23.1 255.255.255.0

ip ospf network point-to-point

exit

ip routing

router ospf 123

router-id 3.3.3.2

auto-cost reference-bandwidth 1000

network 10.10.4.0 0.0.1.255 area 0

network 10.10.16.0 0.0.7.255 area 0

exit

int range G1/0/1 - 10, G1/0/12 - 22, G1/0/24

shut

exit

* + - 1. Save the running configuration to startup-config.

Close configuration window

### Configure PC1 and PC3 with IP addressing.

Configure the two PCs with the IP addresses listed in the Address Table. Also configure their respective default gateways.

### On PC1, verify end-to-end connectivity.

* + - 1. From PC1, **ping** PC3 (i.e., 10.10.5.10).

PC1> **ping 10.10.5.10**

Pinging 10.10.5.10 with 32 bytes of data:

Reply from 10.10.5.10: bytes=32 time=1ms TTL=123

Reply from 10.10.5.10: bytes=32 time=1ms TTL=123

Reply from 10.10.5.10: bytes=32 time=1ms TTL=123

Reply from 10.10.5.10: bytes=32 time=1ms TTL=123

Ping statistics for 10.10.5.10:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 1ms, Maximum = 1ms, Average = 1ms

The pings should be successful. If the pings are unsuccessful, troubleshoot the basic device configurations before continuing.

* + - 1. From PC1, **ping** the first loopback on D3 (i.e., 10.10.16.1).

PC1> **ping 10.10.16.1**

Pinging 10.10.16.1 with 32 bytes of data:

Reply from 10.10.16.1: bytes=32 time=2ms TTL=250

Reply from 10.10.16.1: bytes=32 time=2ms TTL=250

Reply from 10.10.16.1: bytes=32 time=2ms TTL=250

Reply from 10.10.16.1: bytes=32 time=2ms TTL=250

Ping statistics for 10.10.16.1:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 2ms, Maximum = 2ms, Average = 2ms

The pings should be successful. If the pings are unsuccessful, troubleshoot the basic device configurations before continuing.

* + - 1. Finally, from PC1, **ping** the default gateway loopback on R2 (i.e., 209.165.200.225).

PC1> **ping 209.165.200.225**

Pinging 209.165.200.225 with 32 bytes of data:

Reply from 209.165.200.225: bytes=32 time=1ms TTL=253

Reply from 209.165.200.225: bytes=32 time=1ms TTL=253

Reply from 209.165.200.225: bytes=32 time=1ms TTL=253

Reply from 209.165.200.225: bytes=32 time=1ms TTL=253

Ping statistics for 209.165.200.225:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 1ms, Maximum = 1ms, Average = 1ms

The pings should be successful. If the pings are unsuccessful, troubleshoot the basic device configurations before continuing.

### Verify the routing table of R1.

* + - 1. Verify the OSPF routing table of R1.

Open configuration window

R1# **show ip route ospf | begin Gateway**

Gateway of last resort is 64.100.0.1 to network 0.0.0.0

10.0.0.0/8 is variably subnetted, 5 subnets, 3 masks

O 10.10.1.0/24 [110/11] via 10.10.0.2, 00:29:03, GigabitEthernet0/0/1

O 10.10.2.0/24 [110/2] via 10.10.0.2, 00:29:03, GigabitEthernet0/0/1

O 10.10.3.0/24 [110/2] via 10.10.0.2, 00:29:03, GigabitEthernet0/0/1

The routing table confirms that R1 has knowledge of the networks connected to D1. Notice however, that R1 has no knowledge of the routes connected to the R3 OSPF domain. The reason why PC1 can still reach PC3 is because R1 has a default static route to R2. R1 forwarded the traffic to R2 because it did not know where the 10.10.5.0 network was. R2 has a static route to this network and therefore forwarded it to R3.

Close configuration window

## Configure a Site-to-Site VPN using Crypto Maps Between R1 and R3

In Part 2 of this lab, you will configure an IPsec VPN tunnel between R1 and R3 that passes through R2. You will configure R1 and R3 using the Cisco IOS CLI. You will then review and test the resulting configuration.

IPsec is an open framework that allows for the exchange of security protocols as new technologies, and encryption algorithms as they are developed.

There are two central configuration elements in the implementation of an IPsec VPN:

* Implement Internet Key Exchange (IKE) parameters
* Implement IPsec parameters

### On R1 and R3, implement Internet Key Exchange (IKE) parameters.

In this step, you will enable IKE policies on R1 and R3. IKE Phase 1 defines the key exchange method used to exchange and validate IKE policies between peers. In IKE Phase 2, the peers exchange and match IPsec policies for the authentication and encryption of data traffic.

IKE must be enabled for IPsec to function. IKE is enabled, by default, on IOS images with cryptographic feature sets. However, if is disabled, you can enable it with the **crypto isakmp enable** command. This command can also be used to verify that the router IOS supports IKE and that it is enabled.

**Note**: If the command produces an error and cannot be executed, then the device must be upgraded to an IOS image that includes the Cisco cryptographic services.

* + - 1. When a VPN is negotiated, the router will attempt to connect to the other device using the defined ISAKMP policies. To allow IKE Phase 1 negotiation, you must create an ISAKMP policy that defines the authentication, encryption algorithms, and the hash function used to send control traffic between the two VPN endpoints. When an ISAKMP security association has been accepted by the IKE peers, IKE Phase 1 has been completed.

There are multiple ISAKMP policies available by default. On R1, view the ISAKMP policies available using the **show crypto isakmp policy** global config command.

Open configuration window

R1# **show crypto isakmp policy**

Default IKE policy

Protection suite of priority 65507

encryption algorithm: AES - Advanced Encryption Standard (128 bit keys).

hash algorithm: Secure Hash Standard

authentication method: Rivest-Shamir-Adleman Signature

Diffie-Hellman group: #5 (1536 bit)

lifetime: 86400 seconds, no volume limit

Protection suite of priority 65508

encryption algorithm: AES - Advanced Encryption Standard (128 bit keys).

hash algorithm: Secure Hash Standard

authentication method: Pre-Shared Key

Diffie-Hellman group: #5 (1536 bit)

lifetime: 86400 seconds, no volume limit

<Output omitted>

Protection suite of priority 65514

encryption algorithm: Three key triple DES

hash algorithm: Message Digest 5

authentication method: Pre-Shared Key

Diffie-Hellman group: #2 (1024 bit)

lifetime: 86400 seconds, no volume limit

These are default policies that are available if no custom ISAKMP policy is configured. The policies are listed in order of priority with policy 65507 providing the most secure settings and policy 65514 offering the least secure.

Default policies may not provide the required settings for your VPNs. And although default policies exist, it is recommended that you define specific custom policies.

* + - 1. To create a custom ISAKMP policy, enter ISAKMP configuration mode using the **crypto isakmp policy** *number* global configuration mode command. The policy number uniquely identifies the IKE policy and assigns a priority to the policy, where 1 is the highest priority.

On R1, create ISAKMP policy number 10 as shown.

R1(config)# crypto isakmp policy 10

* + - 1. View the various IKE parameters available using Cisco IOS help by typing a question mark (**?**).

R1(config-isakmp)# ?

ISAKMP commands:

authentication Set authentication method for protection suite

default Set a command to its defaults

encryption Set encryption algorithm for protection suite

exit Exit from ISAKMP protection suite configuration mode

group Set the Diffie-Hellman group

hash Set hash algorithm for protection suite

lifetime Set lifetime for ISAKMP security association

no Negate a command or set its defaults

As shown in the table below, several parameters can be configured in ISAKMP policy configuration mode.

| Parameter | Options | Default | Recommended |
| --- | --- | --- | --- |
| authentication | pre-share | rsa-encr | rsa-sig | rsa-sig | (varies) |
| encryption | des | 3des | aes | aes [128 | 192 | 256] | des | aes (or higher) |
| group | 1 | 2 | 5 | 14 | 15 | 16 | 19 | 20 | 24 | 1 | 14 (or higher) |
| hash | md5 | sha | sha256 | sha384 | sha512 | sha | sha256 (or higher) |
| lifetime | 60-86400 (in seconds) | 86400 (24 hours) | (shorter timeframes are more secure) |

**Note**: The last column lists the minimum recommended options.

* + - 1. Entering the ISAKMP policy configuration mode automatically assigns default parameters to the policy. To view these defaults, use the **do show crypto isakmp policy** command.

R1(config-isakmp)# **do show crypto isakmp policy**

Global IKE policy

Protection suite of priority 10

encryption algorithm: DES - Data Encryption Standard (56 bit keys).

hash algorithm: Secure Hash Standard

authentication method: Rivest-Shamir-Adleman Signature

Diffie-Hellman group: #1 (768 bit)

lifetime: 86400 seconds, no volume limit

The output highlights the default parameters automatically assigned to the new policy. For security reason, most of these should be updated to the recommended minimum listed in the table.

Your choice of an encryption algorithm determines how confidential the control channel between the endpoints is. The hash algorithm controls data integrity, ensuring that the data received from a peer has not been tampered with in transit. The authentication type ensures that the packet was sent and signed by the remote peer. The Diffie-Hellman group is used to create a secret key shared by the peers that has not been sent across the network.

* + - 1. In this lab, we will use the following parameters for the ISAKMP policy 10 on R1 and R3:
* Encryption: **aes 256**
* Hash: sha256
* Authentication method: **pre-share key**
* Diffie-Hellman group: **14**
* Lifetime: **3600** seconds (60 minutes / 1 hour)

**Note**: Older versions of Cisco IOS do not support AES 256 encryption and SHA as a hash algorithm. Substitute whatever encryption and hashing algorithm your router supports. Ensure that the same changes are made on R3 in order to be in sync.

R1(config-isakmp)# encryption aes 256

R1(config-isakmp)# hash sha256

R1(config-isakmp)# authentication pre-share

R1(config-isakmp)# group 14

R1(config-isakmp)# lifetime 3600

R1(config-isakmp)# **exit**

close configuration window

* + - 1. Configure the same policy on R3.

Open configuration window

R3(config)# **crypto isakmp policy 10**

R3(config-isakmp)# **encryption aes 256**

R3(config-isakmp)# **hash sha256**

R3(config-isakmp)# **authentication pre-share**

R3(config-isakmp)# **group 14**

R3(config-isakmp)# **lifetime 3600**

close configuration window

* + - 1. Verify the IKE policy with the show crypto isakmp policy command on R1 and R3.

Open configuration window

R1(config)# **do show crypto isakmp policy**

Global IKE policy

Protection suite of priority 10

encryption algorithm: AES - Advanced Encryption Standard (256 bit keys).

hash algorithm: Secure Hash Standard 2 (256 bit)

authentication method: Pre-Shared Key

Diffie-Hellman group: #14 (2048 bit)

lifetime: 3600 seconds, no volume limit

close configuration window

R3(config)# **do show crypto isakmp policy**

close configuration window

Global IKE policy

Protection suite of priority 10

encryption algorithm: AES - Advanced Encryption Standard (256 bit keys).

hash algorithm: Secure Hash Standard 2 (256 bit)

authentication method: Pre-Shared Key

Diffie-Hellman group: #14 (2048 bit)

lifetime: 3600 seconds, no volume limit

The policies must match. Troubleshoot and verify that the configurations were done correctly on both routers.

close configuration window

### On R1 and R3, configure the pre-shared keys.

Because pre-shared keys are used as the authentication method in the IKE policy, a key must be configured on each router that points to the other VPN endpoint. These keys must match for authentication to be successful.

Use the **crypto isakmp key** *key-string* **address** *ip-address* global configuration mode command to enter a pre-shared key. Use the global IP address of the remote peer, which is the outside interface of the remote peer router.

**Note**: The *ip-address* parameter can be assigned**0.0.0.0 0.0.0.0** to allow a match against any peer.

#### Question:

Which IP addresses should you use to configure the IKE peers, given the topology diagram and IP addressing table?

Type your answer here.

The IP addresses should be as follows: R1 G0/0/0 IP address 64.100.0.2 and R3 G0/0/0 IP address 64.100.1.2. These are the addresses that are used to send normal traffic between R1 and R3.

* + - 1. Each IP address that is used to configure the IKE peers is also referred to as the IP address of the remote VPN endpoint. Configure the pre-shared key of **cisco123** on R1. This command points to the remote peer R3 G0/0/0 IP address.

Open configuration window

**Note**: Production networks should use longer and more complex keys.

R1(config)# crypto isakmp key cisco123 address 64.100.1.2

close configuration window

* + - 1. Configure the pre-shared key **cisco123** on router R3. The command for R3 points to the R1 G0/0/0 IP address.

Open configuration window

R3(config)# crypto isakmp key cisco123 address 64.100.0.2

close configuration window

### On R1 and R3, configure the IPsec transform set and lifetime.

The IPsec transform set is another crypto configuration parameter that routers negotiate to form a security association. To create an IPsec transform set, use the **crypto ipsec transform-set** *transform-set-name* *transform1* [*transform2* [*transform3*]]command.

* + - 1. On R1 and R3, create a transform set with the name **S2S-VPN** and use ? to see which parameters are available.

Open configuration window

R1(config)# **crypto ipsec transform-set S2S-VPN ?**

ah-md5-hmac AH-HMAC-MD5 transform

ah-sha-hmac AH-HMAC-SHA transform

ah-sha256-hmac AH-HMAC-SHA256 transform

ah-sha384-hmac AH-HMAC-SHA384 transform

ah-sha512-hmac AH-HMAC-SHA512 transform

esp-192-aes ESP transform using AES cipher (192 bits)

esp-256-aes ESP transform using AES cipher (256 bits)

esp-3des ESP transform using 3DES(EDE) cipher (168 bits)

esp-aes ESP transform using AES cipher

esp-des ESP transform using DES cipher (56 bits)

esp-gcm ESP transform using GCM cipher

esp-gmac ESP transform using GMAC cipher

esp-md5-hmac ESP transform using HMAC-MD5 auth

esp-null ESP transform w/o cipher

esp-seal ESP transform using SEAL cipher (160 bits)

esp-sha-hmac ESP transform using HMAC-SHA auth

esp-sha256-hmac ESP transform using HMAC-SHA256 auth

esp-sha384-hmac ESP transform using HMAC-SHA384 auth

esp-sha512-hmac ESP transform using HMAC-SHA512 auth

R1(config)# **crypto ipsec transform-set S2S-VPN**

* + - 1. On R1 and R3, use the AES 256 cipher with ESP and the SHA 256 hash function.

R1(config)# **crypto ipsec transform-set S2S-VPN esp-aes 256 esp-sha256-hmac**

R1(cfg-crypto-trans)# **exit**

close configuration window

Open configuration window

R3(config)# **crypto ipsec transform-set S2S-VPN esp-aes 256 esp-sha256-hmac**

R3(cfg-crypto-trans)# **exit**

close configuration window

#### Question:

What is the function of the IPsec transform set?

Type your answers here.

The IPsec transform set specifies the cryptographic algorithms and functions (transforms) that a router employs on the actual data packets sent through the IPsec tunnel. These algorithms include the encryption, encapsulation, authentication, and data integrity services that IPsec can apply.

* + - 1. You can also change the IPsec security association lifetime from the default of 3600 seconds. On R1 and R3, set the IPsec security association lifetime to 30 minutes, or 1800 seconds.

Open configuration window

R1(config)# crypto ipsec security-association lifetime seconds 1800

close configuration window

Open configuration window

R3(config)# crypto ipsec security-association lifetime seconds 1800

close configuration window

### On R1 and R3, define interesting traffic.

It is necessary to define interesting traffic that will tell the router to enable an IPsec VPN with the other VPN peer. Do this by using an extended access list to tell the router which traffic to encrypt. A packet that is permitted by an access list used for defining IPsec traffic is encrypted if the IPsec session is configured correctly. A packet that is denied by one of these access lists is not dropped but is instead forwarded to its destination unencrypted. Also, like any other access list, there is an implicit deny at the end, which means the default action is to not encrypt traffic. If there is no IPsec security association correctly configured, no traffic is encrypted and traffic is forwarded unencrypted.

In this scenario, from the perspective of R1, the traffic you want to encrypt is traffic going from the R1 LANs to the R3 Ethernet LANs or vice versa from the perspective of R3. These access lists are used outbound on the VPN endpoint interfaces and must mirror each other.

* + - 1. On R1, identify interesting IPsec VPN traffic using a named extended ACL called **S2S-VPN-ACL**.

Open configuration window

R1(config)# **ip access-list extended S2S-VPN-ACL**

R1(config-ext-nacl)# **remark ACL identifies interesting traffic going to R3**

R1(config-ext-nacl)# **permit ip 10.10.0.0 0.0.3.255 10.10.4.0 0.0.3.255**

R1(config-ext-nacl)# **permit ip 10.10.0.0 0.0.3.255 10.10.16.0 0.0.7.255**

R1(config-ext-nacl)# **exit**

The ACL identifies traffic from the R1 networks going to the R3 networks as interesting.

close configuration window

Open configuration window

* + - 1. Configure the IPsec VPN interesting traffic named extended ACL on R1.

R3(config)# **ip access extended S2S-VPN-ACL**

R3(config-ext-nacl)# **remark ACL identifies interesting traffic going to R1**

R3(config-ext-nacl)# **permit ip 10.10.4.0 0.0.3.255 10.10.0.0 0.0.3.255**

R3(config-ext-nacl)# **permit ip 10.10.16.0 0.0.7.255 10.10.0.0 0.0.3.255**

R3(config-ext-nacl)# **exit**

The ACL identifies traffic from the R3 networks going to the R1 networks as interesting.

close configuration window

#### Question:

Does IPsec evaluate whether the access lists are mirrored as a requirement to negotiate its security association?

Type your answers here

Yes. IPsec does evaluate whether access lists are mirrored. IPsec does not form a security association if the peers do not have mirrored access lists to select interesting traffic.

### On R1 and R3, create and apply a crypto map.

A crypto map associates traffic that matches an access list to a peer and various IKE and IPsec settings. After the crypto map is created, it can be applied to one or more interfaces. The interfaces that it is applied to should be the ones facing the IPsec peer.

To create a crypto map, use **crypto map** *name sequence-number type*command in global configuration mode to enter crypto map configuration mode for that sequence number. Multiple crypto map statements can belong to the same crypto map and are evaluated in ascending numerical order.

* + - 1. Create the crypto map on R1, name it **S2S-CMAP**, use **10** as the sequence number, and set the type as **ipsec-isakmp**, which means IKE is used to establish IPsec security associations.

Open configuration window

R1(config)# **crypto map S2S-CMAP 10 ipsec-isakmp**

% NOTE: This new crypto map will remain disabled until a peer

and a valid access list have been configured.

Notice that a message is displayed after the command is issued.

* + - 1. Use the **match address** ACL command to specify which access list defines which traffic to encrypt.

R1(config-crypto-map)# match address S2S-VPN-ACL

* + - 1. To view the list of possible **set** commands that you can do with a crypto map, use the help function.

R1(config-crypto-map)# **set ?**

identity Identity restriction.

ikev2-profile Specify ikev2 Profile

ip Interface Internet Protocol config commands

isakmp-profile Specify isakmp Profile

nat Set NAT translation

peer Allowed Encryption/Decryption peer.

pfs Specify pfs settings

reverse-route Reverse Route Injection.

security-association Security association parameters

transform-set Specify list of transform sets in priority order

* + - 1. Setting a peer IP or hostname is required. Set it to R3’s remote VPN endpoint interface using the following command:

R1(config-crypto-map)# set peer 64.100.1.2

* + - 1. Use the **set transform-set** *name* command to hard code the transform set to be used with this peer. Set the perfect forwarding secrecy type using the **set pfs** *type* command, and modify the default IPsec security association lifetime with the **set security-association lifetime seconds** *seconds* command.

R1(config-crypto-map)# set pfs group14

R1(config-crypto-map)# **set transform S2S-VPN**

R1(config-crypto-map)# **set security-association lifetime seconds 900**

R1(config-crypto-map)# **exit**

close configuration window

* + - 1. Create a mirrored matching crypto map on R3.

Open configuration window

R3(config)# **crypto map S2S-CMAP 10 ipsec-isakmp**

% NOTE: This new crypto map will remain disabled until a peer

and a valid access list have been configured.

R3(config-crypto-map)# **match address S2S-VPN-ACL**

R3(config-crypto-map)# **set peer 64.100.0.2**

R3(config-crypto-map)# **set pfs group14**

R3(config-crypto-map)# **set transform S2S-VPN**

R3(config-crypto-map)# **set security-association lifetime seconds 900**

R3(config-crypto-map)# **exit**

close configuration window

* + - 1. Apply the crypto map to interfaces.

**Note**: The SAs are not established until the crypto map has been activated by interesting traffic. The router generates a notification that crypto is now on.

Apply the crypto maps to the appropriate interfaces on R1 and R3.

Open configuration window

R1(config)# **interface g0/0/0**

R1(config-if)# **crypto map S2S-CMAP**

R1(config-if)# **end**

\*Jan 29 15:45:20.117: %CRYPTO-6-ISAKMP\_ON\_OFF: ISAKMP is ON

close configuration window

Open configuration window

R3(config)# **interface g0/0/0**

R3(config-if)# **crypto map S2S-CMAP**

R3(config-if)# **end**

\*Jan 29 15:43:29.524: %CRYPTO-6-ISAKMP\_ON\_OFF: ISAKMP is ON

close configuration window

## Verify a Site-to-Site VPN Between R1 and R3

After a VPN is configured, it must be tested to see if it performs as expected.

### Verify the Site-to-Site IPsec VPN Configuration.

Previously, you used the **show crypto isakmp policy** command to display the configured ISAKMP policies on the router.

* + - 1. Use the **show crypto ipsec transform-set** [*transform-set-name*] command to display the configured IPsec policies in the form of the transform sets.

Open configuration window

R1# **show crypto ipsec transform-set S2S-VPN**

{ esp-256-aes esp-sha256-hmac }

will negotiate = { Tunnel, },

The S2S-VPN transform set settings are highlighted in the output.

* + - 1. On R1 and R3, use the **show crypto map** command to display the crypto maps applied to the router.

R1# **show crypto map**

Crypto Map IPv4 "S2S-CMAP" 10 ipsec-isakmp

Peer = 64.100.1.2

Extended IP access list S2S-VPN-ACL

access-list S2S-VPN-ACL permit ip 10.10.0.0 0.0.3.255 10.10.4.0 0.0.3.255

access-list S2S-VPN-ACL permit ip 10.10.0.0 0.0.3.255 10.10.16.0 0.0.7.255

Current peer: 64.100.1.2

Security association lifetime: 4608000 kilobytes/900 seconds

Responder-Only (Y/N): N

PFS (Y/N): Y

DH group: group14

Mixed-mode : Disabled

Transform sets={

S2S-VPN: { esp-256-aes esp-sha256-hmac } ,

}

Interfaces using crypto map S2S-CMAP:

GigabitEthernet0/0/0

close configuration window

Open configuration window

R3# **show crypto map**

Crypto Map IPv4 "S2S-CMAP" 10 ipsec-isakmp

Peer = 64.100.0.2

Extended IP access list S2S-VPN-ACL

access-list S2S-VPN-ACL permit ip 10.10.4.0 0.0.3.255 10.10.0.0 0.0.3.255

access-list S2S-VPN-ACL permit ip 10.10.16.0 0.0.7.255 10.10.0.0 0.0.3.255

Current peer: 64.100.0.2

Security association lifetime: 4608000 kilobytes/900 seconds

Responder-Only (Y/N): N

PFS (Y/N): Y

DH group: group14

Mixed-mode : Disabled

Transform sets={

S2S-VPN: { esp-256-aes esp-sha256-hmac } ,

}

Interfaces using crypto map S2S-CMAP:

GigabitEthernet0/0/0

Note: The output of these **show** commands does not change when the VPN tunnel is enabled.

close configuration window

### Display ISAKMP and IPsec security associations.

* + - 1. The **show crypto isakmp as** command reveals that no IKE SAs exist yet. When interesting traffic is sent, this command output will change.

Open configuration window

R1# show crypto isakmp sa

IPv4 Crypto ISAKMP SA

dst src state conn-id status

IPv6 Crypto ISAKMP SA

* + - 1. The **show crypto ipsec sa** command displays packet statistics information for each of the ACE statements in the VPN ACL. The first portion of the output displays the packet statistics for traffic from the R1 10.10.0.0/22 networks to the R3 10.10.4.0/22 networks. The bottom portion displays the statistics for traffic from the R1 10.10.0.0/22 networks to the R3 10.10.16.0/21 networks.

R1# **show crypto ipsec sa**

interface: GigabitEthernet0/0/0

Crypto map tag: S2S-CMAP, local addr 64.100.0.2

protected vrf: (none)

local ident (addr/mask/prot/port): (10.10.0.0/255.255.252.0/0/0)

remote ident (addr/mask/prot/port): (10.10.4.0/255.255.252.0/0/0)

current\_peer 64.100.1.2 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.: 64.100.0.2, remote crypto endpt.: 64.100.1.2

plaintext mtu 1438, path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0/0/0

current outbound spi: 0x0(0)

PFS (Y/N): N, DH group: none

inbound esp sas:

inbound ah sas:

inbound pcp sas:

outbound esp sas:

outbound ah sas:

outbound pcp sas:

protected vrf: (none)

local ident (addr/mask/prot/port): (10.10.0.0/255.255.252.0/0/0)

remote ident (addr/mask/prot/port): (10.10.16.0/255.255.248.0/0/0)

current\_peer 64.100.1.2 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.: 64.100.0.2, remote crypto endpt.: 64.100.1.2

<Output omitted>

close configuration window

**Note**: The output lists the current status for each ACE in the S2S-VPN-ACL.

#### Question:

Why haven’t any SAs been negotiated?

Type your answers here.

IPsec has not begun to negotiate a SA over which it will encrypt traffic because no interesting traffic has been identified.

* + - 1. Next, we will generate some “uninteresting” test traffic and observe the results. From R1, **ping** the R3 G0/0/0 interface IP address (i.e., 64.100.1.2) and then ping the R3 G0/0/1 interface IP address (i.e., 10.10.4.1). These pings should be successful.
      2. Issue the **show crypto isakmp sa** command again.

#### Question:

Was an SA created for these pings? Explain.

Type your answers here.

No, there are no SAs created. The S2S-VPN-ACL associated with the crypto map for R1 defines interesting traffic as IP packets from R1 LANs to the R3 LANs. The main reason why the SA were not created is because the source of both pings was the R1 G0/0/0 address of 64.100.0.2 which does not match the VPN ACL. Therefore, these pings were not “interesting” traffic.

### Generate some interesting test traffic and observe the results.

A VPN is initiated when interesting traffic is generated. Typically, inside users connecting to the remote network can initiate a VPN connection.

* + - 1. From PC1, **ping** PC3 to generate interesting traffic.

PC1> **ping 10.10.5.10**

Pinging 10.10.5.10 with 32 bytes of data:

Request timed out.

Reply from 10.10.5.10: bytes=32 time=1ms TTL=64

Reply from 10.10.5.10: bytes=32 time=1ms TTL=64

Reply from 10.10.5.10: bytes=32 time=1ms TTL=64

Ping statistics for 10.10.5.10:

Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),

Approximate round trip times in milli-seconds:

Minimum = 1ms, Maximum = 1ms, Average = 1ms

Notice how the first ping reply timed out. The reason is because, the first echo request triggered the S2S-VPN-ACL which made R1 negotiate and establish the IPsec VPN tunnel with R3.

* + - 1. An alternate method to initiate interesting traffic would be to use an extended ping on R1. An extended ping allows you to control the source address of the packets.

R1# **ping 10.10.16.1 source g0/0/1**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.10.16.1, timeout is 2 seconds:

Packet sent with a source address of 10.10.0.1

.!!!!

Success rate is 80 percent (4/5), round-trip min/avg/max = 2/3/4 ms

Like the previous ping, the first echo reply timed out. The reason was to establish an SA for the 10.10.16.0/23 networks.

* + - 1. Re-issue the **show crypto isakmp sa** command.

Open configuration window

R1# **show crypto isakmp sa**

IPv4 Crypto ISAKMP SA

dst src state conn-id status

64.100.1.2 64.100.0.2 QM\_IDLE 1001 ACTIVE

IPv6 Crypto ISAKMP SA

The SA displays that the tunnel is active.

#### Question:

Why was an SA created between R1 and R3 this time?

Type your answers here.

The source of the pings was from the R1 networks to the R3 network. This is interesting traffic based on the ACL S2S-VPN-ACL statements. An SA is established and packets travel through the tunnel as encrypted traffic.

* + - 1. Verify the IPsec traffic statistics using the **show crypto ipsec sa** command.

#### Question:

How many packets have been transformed between R1 and R3?

Type your answers here.

Answers may vary. Seven: Three of the five packets from the R1 to R3 pings, four packets from the PC-A to R3 pings, and one packet for each echo request. The number of packets may vary depending on how many pings have been issued and from where.

R1# **show crypto ipsec sa**

interface: GigabitEthernet0/0/0

Crypto map tag: S2S-CMAP, local addr 64.100.0.2

protected vrf: (none)

local ident (addr/mask/prot/port): (10.10.0.0/255.255.252.0/0/0)

remote ident (addr/mask/prot/port): (10.10.4.0/255.255.252.0/0/0)

current\_peer 64.100.1.2 port 500

PERMIT, flags={origin\_is\_acl,}

#pkts encaps: 3, #pkts encrypt: 3, #pkts digest: 3

#pkts decaps: 3, #pkts decrypt: 3, #pkts verify: 3

#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.: 64.100.0.2, remote crypto endpt.: 64.100.1.2

plaintext mtu 1438, path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0/0/0

current outbound spi: 0xD225EBA7(3525700519)

PFS (Y/N): Y, DH group: group14

inbound esp sas:

spi: 0xB2682427(2993169447)

transform: esp-256-aes esp-sha256-hmac ,

in use settings ={Tunnel, }

conn id: 2007, flow\_id: ESG:7, sibling\_flags FFFFFFFF80004048, crypto map: S2S-CMAP

sa timing: remaining key lifetime (k/sec): (4608000/590)

IV size: 16 bytes

replay detection support: Y

Status: ACTIVE(ACTIVE)

inbound ah sas:

inbound pcp sas:

outbound esp sas:

spi: 0xD225EBA7(3525700519)

transform: esp-256-aes esp-sha256-hmac ,

in use settings ={Tunnel, }

conn id: 2008, flow\_id: ESG:8, sibling\_flags FFFFFFFF80004048, crypto map: S2S-CMAP

sa timing: remaining key lifetime (k/sec): (4608000/590)

IV size: 16 bytes

replay detection support: Y

Status: ACTIVE(ACTIVE)

outbound ah sas:

outbound pcp sas:

protected vrf: (none)

local ident (addr/mask/prot/port): (10.10.0.0/255.255.252.0/0/0)

remote ident (addr/mask/prot/port): (10.10.16.0/255.255.248.0/0/0)

current\_peer 64.100.1.2 port 500

PERMIT, flags={origin\_is\_acl,}

#pkts encaps: 4, #pkts encrypt: 4, #pkts digest: 4

#pkts decaps: 4, #pkts decrypt: 4, #pkts verify: 4

#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

<Output omitted>

close configuration window

* + - 1. The previous example used pings to generate interesting traffic.

#### Questions:

What other types of traffic would result in an SA forming and tunnel establishment?

Type your answers here.

Any traffic (e.g., FTP, HTTP, Telnet, and others) initiated from an R1 network with a destination address to the R3 networks would be interesting traffic.

Are routing protocols able to traverse an IPsec site-to-site VPN?

Type your answers here.

Not by default. GRE would need to be configured with IPsec to support routing protocols between sites.

# Router Interface Summary Table

| Router Model | Ethernet Interface #1 | Ethernet Interface #2 | Serial Interface #1 | Serial Interface #2 |
| --- | --- | --- | --- | --- |
| 1800 | Fast Ethernet 0/0 (F0/0) | Fast Ethernet 0/1 (F0/1) | Serial 0/0/0 (S0/0/0) | Serial 0/0/1 (S0/0/1) |
| 1900 | Gigabit Ethernet 0/0 (G0/0) | Gigabit Ethernet 0/1 (G0/1) | Serial 0/0/0 (S0/0/0) | Serial 0/0/1 (S0/0/1) |
| 2801 | Fast Ethernet 0/0 (F0/0) | Fast Ethernet 0/1 (F0/1) | Serial 0/1/0 (S0/1/0) | Serial 0/1/1 (S0/1/1) |
| 2811 | Fast Ethernet 0/0 (F0/0) | Fast Ethernet 0/1 (F0/1) | Serial 0/0/0 (S0/0/0) | Serial 0/0/1 (S0/0/1) |
| 2900 | Gigabit Ethernet 0/0 (G0/0) | Gigabit Ethernet 0/1 (G0/1) | Serial 0/0/0 (S0/0/0) | Serial 0/0/1 (S0/0/1) |
| 4221 | Gigabit Ethernet 0/0/0 (G0/0/0) | Gigabit Ethernet 0/0/1 (G0/0/1) | Serial 0/1/0 (S0/1/0) | Serial 0/1/1 (S0/1/1) |
| 4300 | Gigabit Ethernet 0/0/0 (G0/0/0) | Gigabit Ethernet 0/0/1 (G0/0/1) | Serial 0/1/0 (S0/1/0) | Serial 0/1/1 (S0/1/1) |

**Note**: To find out how the router is configured, look at the interfaces to identify the type of router and how many interfaces the router has. There is no way to effectively list all the combinations of configurations for each router class. This table includes identifiers for the possible combinations of Ethernet and Serial interfaces in the device. The table does not include any other type of interface, even though a specific router may contain one. An example of this might be an ISDN BRI interface. The string in parenthesis is the legal abbreviation that can be used in Cisco IOS commands to represent the interface.

End of document

# Device Configs – Final

# Router R1

R1# **show running-config**

Building configuration...

Current configuration : 2141 bytes

!

version 16.9

service timestamps debug datetime msec

service timestamps log datetime msec

platform qfp utilization monitor load 80

no platform punt-keepalive disable-kernel-core

!

hostname R1

!

boot-start-marker

boot-end-marker

!

no aaa new-model

!

no ip domain lookup

!

login on-success log

!

subscriber templating

!

multilink bundle-name authenticated

!

spanning-tree extend system-id

!

redundancy

mode none

!

crypto isakmp policy 10

encr aes 256

hash sha256

authentication pre-share

group 14

lifetime 3600

crypto isakmp key cisco123 address 64.100.1.2

!

crypto ipsec security-association lifetime seconds 1800

!

crypto ipsec transform-set S2S-VPN esp-aes 256 esp-sha256-hmac

mode tunnel

!

crypto map S2S-CMAP 10 ipsec-isakmp

set peer 64.100.1.2

set security-association lifetime seconds 900

set transform-set S2S-VPN

set pfs group14

match address S2S-VPN-ACL

!

interface GigabitEthernet0/0/0

description Connection to R2

ip address 64.100.0.2 255.255.255.252

negotiation auto

crypto map S2S-CMAP

!

interface GigabitEthernet0/0/1

description Connection to D1

ip address 10.10.0.1 255.255.255.252

negotiation auto

!

interface Serial0/1/0

no ip address

!

interface Serial0/1/1

no ip address

!

router ospf 123

router-id 1.1.1.1

auto-cost reference-bandwidth 1000

network 10.10.0.0 0.0.0.3 area 0

default-information originate

!

ip forward-protocol nd

no ip http server

ip http secure-server

ip route 0.0.0.0 0.0.0.0 64.100.0.1

!

ip access-list extended S2S-VPN-ACL

remark ACL identifies interesting traffic going to R3

permit ip 10.10.0.0 0.0.3.255 10.10.4.0 0.0.3.255

permit ip 10.10.0.0 0.0.3.255 10.10.16.0 0.0.7.255

!

control-plane

!

banner motd ^C This is R1, Implement IPsec Site-to-Site VPNs ^C

!

line con 0

exec-timeout 0 0

logging synchronous

transport input none

stopbits 1

line aux 0

stopbits 1

line vty 0 4

login

!

end

# Router R2

R2# **show running-config**

Building configuration...

Current configuration : 1478 bytes

!

version 16.9

service timestamps debug datetime msec

service timestamps log datetime msec

platform qfp utilization monitor load 80

no platform punt-keepalive disable-kernel-core

!

hostname R2

!

boot-start-marker

boot-end-marker

!

no aaa new-model

!

no ip domain lookup

!

login on-success log

!

subscriber templating

!

multilink bundle-name authenticated

!

spanning-tree extend system-id

!

redundancy

mode none

!

interface Loopback0

description Internet simulated address

ip address 209.165.200.225 255.255.255.224

!

interface GigabitEthernet0/0/0

description Connection to R1

ip address 64.100.0.1 255.255.255.252

negotiation auto

!

interface GigabitEthernet0/0/1

description Connection to R3

ip address 64.100.1.1 255.255.255.252

negotiation auto

!

ip forward-protocol nd

no ip http server

ip http secure-server

ip route 0.0.0.0 0.0.0.0 Loopback0

ip route 10.10.0.0 255.255.252.0 64.100.0.2

ip route 10.10.4.0 255.255.252.0 64.100.1.2

ip route 10.10.16.0 255.255.248.0 64.100.1.2

!

control-plane

!

banner motd ^C This is R2, Implement IPsec Site-to-Site VPNs ^C

!

line con 0

exec-timeout 0 0

logging synchronous

transport input none

stopbits 1

line aux 0

stopbits 1

line vty 0 4

login

!

end

# Router R3

R3# **show running-config**

Current configuration : 2141 bytes

!

version 16.9

service timestamps debug datetime msec

service timestamps log datetime msec

platform qfp utilization monitor load 80

no platform punt-keepalive disable-kernel-core

!

hostname R3

!

boot-start-marker

boot-end-marker

!

no aaa new-model

!

no ip domain lookup

!

login on-success log

!

subscriber templating

!

multilink bundle-name authenticated

!

spanning-tree extend system-id

!

redundancy

mode none

!

crypto isakmp policy 10

encr aes 256

hash sha256

authentication pre-share

group 14

lifetime 3600

crypto isakmp key cisco123 address 64.100.0.2

!

crypto ipsec security-association lifetime seconds 1800

!

crypto ipsec transform-set S2S-VPN esp-aes 256 esp-sha256-hmac

mode tunnel

!

crypto map S2S-CMAP 10 ipsec-isakmp

set peer 64.100.0.2

set security-association lifetime seconds 900

set transform-set S2S-VPN

set pfs group14

match address S2S-VPN-ACL

!

interface GigabitEthernet0/0/0

description Connection to R2

ip address 64.100.1.2 255.255.255.252

negotiation auto

crypto map S2S-CMAP

!

interface GigabitEthernet0/0/1

description Connection to D3

ip address 10.10.4.1 255.255.255.252

negotiation auto

!

interface Serial0/1/0

no ip address

!

interface Serial0/1/1

no ip address

!

router ospf 123

router-id 3.3.3.1

auto-cost reference-bandwidth 1000

network 10.10.4.0 0.0.0.3 area 0

default-information originate

!

ip forward-protocol nd

no ip http server

ip http secure-server

ip route 0.0.0.0 0.0.0.0 64.100.1.1

!

ip access-list extended S2S-VPN-ACL

remark ACL identifies interesting traffic going to R1

permit ip 10.10.4.0 0.0.3.255 10.10.0.0 0.0.3.255

permit ip 10.10.16.0 0.0.7.255 10.10.0.0 0.0.3.255

!

control-plane

!

banner motd ^C This is R3, Implement IPsec Site-to-Site VPNs ^C

!

line con 0

exec-timeout 0 0

logging synchronous

transport input none

stopbits 1

line aux 0

stopbits 1

line vty 0 4

login

!

end

# Layer 3 Switch D1

D1# **show running-config**

Building configuration...

Current configuration : 7031 bytes

!

version 16.9

no service pad

service timestamps debug datetime msec

service timestamps log datetime msec

! Call-home is enabled by Smart-Licensing.

service call-home

no platform punt-keepalive disable-kernel-core

!

hostname D1

!

vrf definition Mgmt-vrf

!

address-family ipv4

exit-address-family

!

address-family ipv6

exit-address-family

!

no aaa new-model

switch 1 provision ws-c3650-24ps

!

ip routing

!

no ip domain lookup

!

login on-success log

!

crypto pki trustpoint SLA-TrustPoint

enrollment pkcs12

revocation-check crl

!

license boot level ipservicesk9

!

diagnostic bootup level minimal

!

spanning-tree mode rapid-pvst

spanning-tree extend system-id

!

redundancy

mode sso

!

transceiver type all

monitoring

!

class-map match-any system-cpp-police-topology-control

description Topology control

class-map match-any system-cpp-police-sw-forward

description Sw forwarding, L2 LVX data, LOGGING

class-map match-any system-cpp-default

description Inter FED, EWLC control, EWLC data

class-map match-any system-cpp-police-sys-data

description Learning cache ovfl, High Rate App, Exception, EGR Exception, NFLSAMPLED DATA, RPF Failed

class-map match-any system-cpp-police-punt-webauth

description Punt Webauth

class-map match-any system-cpp-police-l2lvx-control

description L2 LVX control packets

class-map match-any system-cpp-police-forus

description Forus Address resolution and Forus traffic

class-map match-any system-cpp-police-multicast-end-station

description MCAST END STATION

class-map match-any system-cpp-police-multicast

description Transit Traffic and MCAST Data

class-map match-any system-cpp-police-l2-control

description L2 control

class-map match-any system-cpp-police-dot1x-auth

description DOT1X Auth

class-map match-any system-cpp-police-data

description ICMP redirect, ICMP\_GEN and BROADCAST

class-map match-any system-cpp-police-stackwise-virt-control

description Stackwise Virtual

class-map match-any non-client-nrt-class

class-map match-any system-cpp-police-routing-control

description Routing control and Low Latency

class-map match-any system-cpp-police-protocol-snooping

description Protocol snooping

class-map match-any system-cpp-police-dhcp-snooping

description DHCP snooping

class-map match-any system-cpp-police-system-critical

description System Critical and Gold Pkt

!

policy-map system-cpp-policy

!

interface Loopback2

description Loopback to simulate an OSPF network

ip address 10.10.2.1 255.255.255.0

ip ospf network point-to-point

!

interface Loopback3

description Loopback to simulate an OSPF network

ip address 10.10.3.1 255.255.255.0

ip ospf network point-to-point

!

interface GigabitEthernet0/0

vrf forwarding Mgmt-vrf

no ip address

negotiation auto

!

interface GigabitEthernet1/0/1

shutdown

!

interface GigabitEthernet1/0/2

shutdown

!

interface GigabitEthernet1/0/3

shutdown

!

interface GigabitEthernet1/0/4

shutdown

!

interface GigabitEthernet1/0/5

shutdown

!

interface GigabitEthernet1/0/6

shutdown

!

interface GigabitEthernet1/0/7

shutdown

!

interface GigabitEthernet1/0/8

shutdown

!

interface GigabitEthernet1/0/9

shutdown

!

interface GigabitEthernet1/0/10

shutdown

!

interface GigabitEthernet1/0/11

description Connection to R1

no switchport

ip address 10.10.0.2 255.255.255.252

!

interface GigabitEthernet1/0/12

shutdown

!

interface GigabitEthernet1/0/13

shutdown

!

interface GigabitEthernet1/0/14

shutdown

!

interface GigabitEthernet1/0/15

shutdown

!

interface GigabitEthernet1/0/16

shutdown

!

interface GigabitEthernet1/0/17

shutdown

!

interface GigabitEthernet1/0/18

shutdown

!

interface GigabitEthernet1/0/19

shutdown

!

interface GigabitEthernet1/0/20

shutdown

!

interface GigabitEthernet1/0/21

shutdown

!

interface GigabitEthernet1/0/22

shutdown

!

interface GigabitEthernet1/0/23

description Connection to PC1

no switchport

ip address 10.10.1.1 255.255.255.0

!

interface GigabitEthernet1/0/24

shutdown

!

interface GigabitEthernet1/1/1

!

interface GigabitEthernet1/1/2

!

interface GigabitEthernet1/1/3

!

interface GigabitEthernet1/1/4

!

interface Vlan1

no ip address

!

router ospf 123

router-id 1.1.1.2

auto-cost reference-bandwidth 1000

network 10.10.0.0 0.0.3.255 area 0

!

ip forward-protocol nd

ip http server

ip http secure-server

!

control-plane

service-policy input system-cpp-policy

!

banner motd ^C This is D1, Implement IPsec Site-to-Site VPNs ^C

!

line con 0

exec-timeout 0 0

logging synchronous

stopbits 1

line aux 0

stopbits 1

line vty 0 4

login

line vty 5 15

login

!

end

# Layer 3 Switch D3

D3# **show running-config**

Building configuration...

Current configuration : 7924 bytes

!

version 16.9

no service pad

service timestamps debug datetime msec

service timestamps log datetime msec

! Call-home is enabled by Smart-Licensing.

service call-home

no platform punt-keepalive disable-kernel-core

!

hostname D3

!

vrf definition Mgmt-vrf

!

address-family ipv4

exit-address-family

!

address-family ipv6

exit-address-family

!

no aaa new-model

switch 1 provision ws-c3650-24ps

!

ip routing

!

no ip domain lookup

!

login on-success log

!

license boot level ipservicesk9

!

diagnostic bootup level minimal

!

spanning-tree mode rapid-pvst

spanning-tree extend system-id

!

redundancy

mode sso

!

transceiver type all

monitoring

!

class-map match-any system-cpp-police-topology-control

description Topology control

class-map match-any system-cpp-police-sw-forward

description Sw forwarding, L2 LVX data, LOGGING

class-map match-any system-cpp-default

description Inter FED, EWLC control, EWLC data

class-map match-any system-cpp-police-sys-data

description Learning cache ovfl, High Rate App, Exception, EGR Exception, NFLSAMPLED DATA, RPF Failed

class-map match-any system-cpp-police-punt-webauth

description Punt Webauth

class-map match-any system-cpp-police-l2lvx-control

description L2 LVX control packets

class-map match-any system-cpp-police-forus

description Forus Address resolution and Forus traffic

class-map match-any system-cpp-police-multicast-end-station

description MCAST END STATION

class-map match-any system-cpp-police-multicast

description Transit Traffic and MCAST Data

class-map match-any system-cpp-police-l2-control

description L2 control

class-map match-any system-cpp-police-dot1x-auth

description DOT1X Auth

class-map match-any system-cpp-police-data

description ICMP redirect, ICMP\_GEN and BROADCAST

class-map match-any system-cpp-police-stackwise-virt-control

description Stackwise Virtual

class-map match-any non-client-nrt-class

class-map match-any system-cpp-police-routing-control

description Routing control and Low Latency

class-map match-any system-cpp-police-protocol-snooping

description Protocol snooping

class-map match-any system-cpp-police-dhcp-snooping

description DHCP snooping

class-map match-any system-cpp-police-system-critical

description System Critical and Gold Pkt

!

policy-map system-cpp-policy

!

interface Loopback16

description Loopback to simulate an OSPF network

ip address 10.10.16.1 255.255.255.0

ip ospf network point-to-point

!

interface Loopback17

description Loopback to simulate an OSPF network

ip address 10.10.17.1 255.255.255.0

ip ospf network point-to-point

!

interface Loopback18

description Loopback to simulate an OSPF network

ip address 10.10.18.1 255.255.255.0

ip ospf network point-to-point

!

interface Loopback19

description Loopback to simulate an OSPF network

ip address 10.10.19.1 255.255.255.0

ip ospf network point-to-point

!

interface Loopback20

description Loopback to simulate an OSPF network

ip address 10.10.20.1 255.255.255.0

ip ospf network point-to-point

!

interface Loopback21

description Loopback to simulate an OSPF network

ip address 10.10.21.1 255.255.255.0

ip ospf network point-to-point

!

interface Loopback22

description Loopback to simulate an OSPF network

ip address 10.10.22.1 255.255.255.0

ip ospf network point-to-point

!

interface Loopback23

description Loopback to simulate an OSPF network

ip address 10.10.23.1 255.255.255.0

ip ospf network point-to-point

!

interface GigabitEthernet0/0

vrf forwarding Mgmt-vrf

no ip address

negotiation auto

!

interface GigabitEthernet1/0/1

shutdown

!

interface GigabitEthernet1/0/2

shutdown

!

interface GigabitEthernet1/0/3

shutdown

!

interface GigabitEthernet1/0/4

shutdown

!

interface GigabitEthernet1/0/5

shutdown

!

interface GigabitEthernet1/0/6

shutdown

!

interface GigabitEthernet1/0/7

shutdown

!

interface GigabitEthernet1/0/8

shutdown

!

interface GigabitEthernet1/0/9

shutdown

!

interface GigabitEthernet1/0/10

shutdown

!

interface GigabitEthernet1/0/11

description Connection to R3

no switchport

ip address 10.10.4.2 255.255.255.252

!

interface GigabitEthernet1/0/12

shutdown

!

interface GigabitEthernet1/0/13

shutdown

!

interface GigabitEthernet1/0/14

shutdown

!

interface GigabitEthernet1/0/15

shutdown

!

interface GigabitEthernet1/0/16

shutdown

!

interface GigabitEthernet1/0/17

shutdown

!

interface GigabitEthernet1/0/18

shutdown

!

interface GigabitEthernet1/0/19

shutdown

!

interface GigabitEthernet1/0/20

shutdown

!

interface GigabitEthernet1/0/21

shutdown

!

interface GigabitEthernet1/0/22

shutdown

!

interface GigabitEthernet1/0/23

description Connection to PC3

no switchport

ip address 10.10.5.1 255.255.255.0

!

interface GigabitEthernet1/0/24

shutdown

!

interface GigabitEthernet1/1/1

!

interface GigabitEthernet1/1/2

!

interface GigabitEthernet1/1/3

!

interface GigabitEthernet1/1/4

!

interface Vlan1

no ip address

!

router ospf 123

router-id 3.3.3.2

auto-cost reference-bandwidth 1000

network 10.10.4.0 0.0.1.255 area 0

network 10.10.16.0 0.0.7.255 area 0

!

ip forward-protocol nd

ip http server

ip http secure-server

!

control-plane

service-policy input system-cpp-policy

!

banner motd ^C This is D3, Implement IPsec Site-to-Site VPNs ^C

!

line con 0

exec-timeout 0 0

logging synchronous

stopbits 1

line aux 0

stopbits 1

line vty 0 4

login

line vty 5 15

login

!

end