IPSec Lab

In this lab we will configure IPSec tunnels between pairs of access routers. We will ensure that the traffic for one pair of IPv4 and one pair of IPv6 prefixes is encrypted while traffic between another pair of prefixes is unencrypted. This will allow us to see the difference.

We will be configuring the tunnels between

- A1 and A2
- A3 and A4
- A5 and A6

In the rest of these examples you can take X to mean the number of your group and Y to mean the number of the group you are building the VPN with.

Set up a secondary addresses on the Loopback Interface

We needs to have some addresses which we can ping. The loopback interface gives us a place to configure these.

```
interface Loopback 0
ip address 100.68.X0.65 255.255.255.255 secondary
ip address 100.68.X0.128 255.255.255.255 secondary
ipv6 address 2001:DB8:X0:4000::1/128
ipv6 address 2001:DB8:X0:5000::1/128
```

Set up a second customer network on the Access Router

We need to configure a second access network. We will see why later

```
router bgp 10X
 address-family ipv4
  network 100.68.X0.128 mask 255.255.255.192
 address-family ipv6
  network 2001:DB8:X0:5000::/52
!
ip route 100.68.X0.128 255.255.255.192 Null0
ipv6 route 2001:DB8:X0:5000::/52 Null0
```

Configure access-list

To match the interesting traffic to encrypt first step is to define the access-list specifying the source

and destination from both side on the tunnel. You will need to configure one access list for IPv4 traffic and one for IPv6 Here is an example access-list for CPE router R13 (Need for both IPv6 and IPv4):

```
ip access-list extended MATCH-IPv4
 permit ip 100.68.X0.64 0.0.0.63 100.68.Y0.64 0.0.0.63
```

```
ipv6 access-list MATCH-IPv6
 permit 2001:DB8:X0:4000::/52 2001:DB8:Y0:4000::/52
```

Create ISAKMP Policy

Second step to setup an IPSec tunnel is to configuration ISAKMP policy parameters. There are five policy parameters need to be defined to each policy entry. Here are those parameters and their default values:

1. IKE policy encryption: Data Encryption Standard (DES) as the default,
2. IKE policy hash: Secure Hash Standard-1 (SHA-1) as the default,
3. IKE key exchange: Diffie-Hellman Group 1 (768-Bit) as the default,
4. IKE lifetime: One-day (86,400 seconds) lifetime as the default,
5. IKE authentication: RSA public key as the default.

Below is an example of an ISAKMP policy configuration for one of the CPE router:

```
crypto isakmp policy 1
 encr aes
 authentication pre-share
 group 5
```

Create Pre Shared Key: There are three methods can be used for peer authentication in IPSec VPN. I.e.:

1. Pre-shared keys: A secret key configured into each peer manually by the administrator
2. RSA signature: Digital certificate exchanged among the per to authenticate.
3. RSA encrypted nonces: An encrypted random number generated by each IPSec peer then exchanged to authenticate. Two nonces are use during the authentication process.

We will be using a symmetric key, which is pre-shared and need to be shared between IPSec peers out of band. Please note the key command below and address is your tunnel destination which is the WAN address of your peer You need to replace the address with your peer WAN address. You may need to connect to ask another team to find the WAN address you need to configure the key for. Here is an example pre-shared key configuration on R13 (You will need to configure keys for both IPv6 and IPv4):

```
crypto isakmp key Tr@ining123 address 100.68.Y0.26
crypto isakmp key Tr@ining123 address ipv6 2001:DB8:Y0:12::1/128
```

Configure IPSec transform set

IPSec transform sets are exchange between peers during quick mode in phase 2. A transform set is a combination of algorithms and protocol that endorse a security policy for traffic. Below is an example transform set configuration for one of the CPE router:

```
crypto ipsec transform-set ESP-AES-SHA esp-aes esp-sha-hmac
 mode tunnel
```

Creating the crypto map

Now need to create a crypto map to glue all those policy together. Please note the peer address which will be your IPSec tunnel destination. Normally WAN address of remote peer. Below is an example crypto map configuration for one of the CPE router (You will need two crypto maps, one for IPv6 and one for IPv4):

```
crypto map IPv4-LAB-VPN 10 ipsec-isakmp
 set peer 100.68.Y0.26
 set transform-set ESP-AES-SHA
 match address MATCH-IPv4
```

```
crypto map ipv6 IPv6-LAB-VPN 10 ipsec-isakmp
 set peer 2001:DB8:Y0:12::1
 set transform-set ESP-AES-SHA
 match address MATCH-IPv6
```

Apply crypto map to an interface

The final step is to apply the crypto map to an outgoing interface. Below is an example crypto map configuration for one of the CPE router (Don't forget that you wil need to apply maps for both IPv6 and IPv4:

```
interface FastEth 0/0
 ipv6 crypto map IPv6-LAB-VPN
 crypto map IPv4-LAB-VPN
```

Verify your IPSec configuration

Command to show ISAKMP security associations (SAs) built between peers:

```
show crypto isakmp sa
```

Command to shows IPsec SAs built between peers:

```
show crypto ipsec sa
```

Command to verify ISAKMP peer:

```
 show crypto isakmp peers
```

Testing with ping

```
  #ping
  Protocol [ip]: ipv6
  Target IPv6 address: 2001:DB8:Y0:4000::1
  Repeat count [5]:
  Datagram size [100]:
  Timeout in seconds [2]:
  Extended commands? [no]: y
  Source address or interface: 2001:DB8:X0:4000::1
  UDP protocol? [no]:
  Verbose? [no]:
  Precedence [0]:
  DSCP [0]:
  Include hop by hop option? [no]:
  Include destination option? [no]:
  Sweep range of sizes? [no]:
  Type escape sequence to abort.
  Sending 5, 100-byte ICMP Echos to 2001:DB8:Y0:4000::1, timeout is 2
seconds:
  Packet sent with a source address of 2001:DB8:X0:4000::1
  .!!!!
```

We can see if the packets were encrypted with IPSec by looking at the IPSec SA

```
  #show crypto ipsec sa
```

```
  interface: FastEthernet0/0
      Crypto map tag: IPv6-LAB-VPN, local addr 2001:DB8:10:12::1
```

```
     protected vrf: (none)
     local  ident (addr/mask/prot/port): (2001:DB8:10:4000::/52/0/0)
     remote ident (addr/mask/prot/port): (2001:DB8:20:4000::/52/0/0)
     current_peer 2001:DB8:20:12::1 port 500
       PERMIT, flags={origin_is_acl,}
      #pkts encaps: 73, #pkts encrypt: 73, #pkts digest: 73
      #pkts decaps: 73, #pkts decrypt: 73, #pkts verify: 73
      #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
```

Here we can see that 73 packets have been encrypted with this SA. Run the ping again to confirm that this number increases

This is what a network capture between two hosts looks like

Run a ping between the second access network. What do you notice about the IPSec SA counters this time?

```
  #ping
  Protocol [ip]: ipv6
  Target IPv6 address: 2001:DB8:20:5000::1
  Repeat count [5]:
  Datagram size [100]:
  Timeout in seconds [2]:
  Extended commands? [no]: y
  Source address or interface: 2001:DB8:10:5000::1
  UDP protocol? [no]:
  Verbose? [no]:
  Precedence [0]:
  DSCP [0]:
  Include hop by hop option? [no]:
  Include destination option? [no]:
  Sweep range of sizes? [no]:
  Type escape sequence to abort.
  Sending 5, 100-byte ICMP Echos to 2001:DB8:20:5000::1, timeout is 2
seconds:
  Packet sent with a source address of 2001:DB8:10:5000::1
  !!!!!
  Success rate is 100 percent (5/5), round-trip min/avg/max = 72/74/76 ms
```

This is what a network capture between the hosts looks like this time