IPSec Troubleshooting: Problem Scenarios Part 1:--

VPN Introduction:

VPN tunnels are used to connect physically isolated networks that are more often than not separated by nonsecure internetworks. To protect these connections, we employ the IP Security (IPSec) protocol to make secure the transmission of data, voice, and video between sites. These secure tunnels over the Internet public network are encrypted using a number of advanced algorithms to provide confidentiality of data that is transmitted between multiple sites

Encryption will be provided by IPSec in concert with VPN tunnels. The Internet Security Association and Key Management Protocol (ISAKMP) and IPSec are essential to building and encrypting VPN tunnels. ISAKMP, also called IKE (Internet Key Exchange), is the negotiation protocol that allows hosts to agree on how to build an IPSec security association.

ISAKMP negotiation consists of two phases:

- Phase 1 creates the first tunnel, which protects later ISAKMP negotiation messages.
- Phase 2 creates the tunnel that protects data.

IPSec then encrypts exchanged data by employing encryption algorithms that result in authentication, encryption, and critical anti-replay services.

It provides -

Confidentiality(Encrypt Data),
Integrity(At each end of tunnel, Checksum or Hash value is calculated)
and Authenticity (Using Signature and Certificates)

ISAKMP States in ASA :

MM_WAIT_MSG2 : Initial DH public key sent to responder. Awating initial contact reply from other side. if stuck here it usually mean the other end is not responding. This could be due to no route to the far end does not have isakmp enabled on the outside or the far end is down.

MM_WAIT_MSG3 : Both peers have agreeded on the ISAKMP policies. Awating exchange of keying information. Hang up here may be due to mismatch device vendors, a router with a firewall in the way or even ASA version mismatch.

MM_WAIT_MSG4 : In this step the pre-share key hashes are exchanged. They are not compared or checked, only sent.If one side sends a key and does not receive a key back, this is where the tunnel will fail. I have seen the tunnel fail at this step due to the remote side having wrong peer ip address due to the remote side having wrong peer ip address. Hang up here may also be due to mismatch device vendors, a router with a firewall in the way or even ASA version mismatch.

MM_WAIT_MSG5 : This step is where the device exchange pre-shareed keys. If the pre-shared keys do not match it will stay at this MSG. I have also seen the tunnel stop here when NAT traversal was on when it needed to be turned off.

MM_WAIT_MSG6 : This step is where the device exchange pre-shared keys. IF the pre-shared keys do not match it will stay at this MSG. I have also seen the tunnel stop here when NAT traversal was on when it needed to be turned off. However if the state goes to MSG6 then the isakmp gets reset that means phase 1 finishes but phase 2 failed. Check that ipsec setting match in phase 2 to get the tunnel to
MM_ACTIVE.

MM_ACTIVE : This isakmp negotiation are complete. Phase 1 has sucessfully completed.

####################

ISAKMP States: For Router

State	Explanation
MM_NO_STATE	The ISAKMP SA has been created, but nothing else has happened yet. It is "larval" at this stage.
MM_SA_SETUP	The peers have agreed upon parameters for the ISAKMP SA
MM_KEY_EXCH	The peers have exchanged Diffie-Hellman public kets and have generated a shared secret. The ISAKMP SA remains unauthenticated.
MM_KEY_AUTH	The ISAKMP SA has been authenticated. If the router initiated this exchange, this states transitions immediately to QM_IDLE, and a Quick Mode exchange begins.
QM_IDLE	The ISAKMP SA is Idle. Will show this state after successful Phase 1 negotiation

Main Mode vs. Aggressive Mode:-

There are two phases of the IKE negotiations, called Phase 1 and Phase 2. Phase 1 can be configured to use either Main Mode or Aggressive Mode. Main Mode is more secure in providing identity protection for the negotiating nodes. However, Main Mode requires a static IP address on both IPSec security devices negotiating the VPN tunnel.

Aggressive Mode is used when one IPSec security device has a dynamic WAN IP address (i.e., uses DHCP, PPPoE, PPPoA, PPTP, etc.). Aggressive Mode has more configuration requirements than Main Mode and may be difficult or impossible to achieve with some IPSec security device pairings.

To configure IKE Phase 1, you need to configure ISAKMP policies. It is possible to configure multiple policies with different configuration statements and then let the two hosts negotiate the policies

Main Mode: MM_NO_STATE , MM_SA_SETUP, MM_KEY_EXCH and MM_KEY_AUTH

Aggressive Mode: AG_NO_STATE ,AG_INIT_EXCH and AG_AUTH .

- **IKE Phase 1**: The two ISAKMP peers establish a secure and an authenticated channel. This channel is known as the ISAKMP SA. There are two modes defined by ISAKMP: Main Mode(Default) and Aggressive Mode.

Aggressive mode : : Use three-way packet exchange to establish tunnel, Fast but not secure

Main mode: Use six-way packet exchange to establish tunnel, slow but secure.It is a Default mode.

- **IKE Phase 2**: SAs are negotiated on behalf of services such as IPSec that need keying material. This phase is called Quick Mode.

IKE VPN Protocols : The IKE protocol is used during the entire negotiation phase. The negotiation defines policy settings and keys used by the IPSec tunnel protocol. The protocols used for the IKE negotiation and VPN tunnel are as follows:

Standard

- TCP port 50 for IPSec Encapsulating Security Protocol (ESP) traffic
- TCP port 51 for IPSec Authentication Header (AH) traffic
- UDP port 500 for Internet Key Exchange (IKE) negotiation traffic

With NAT Traversal (NAT-T) active

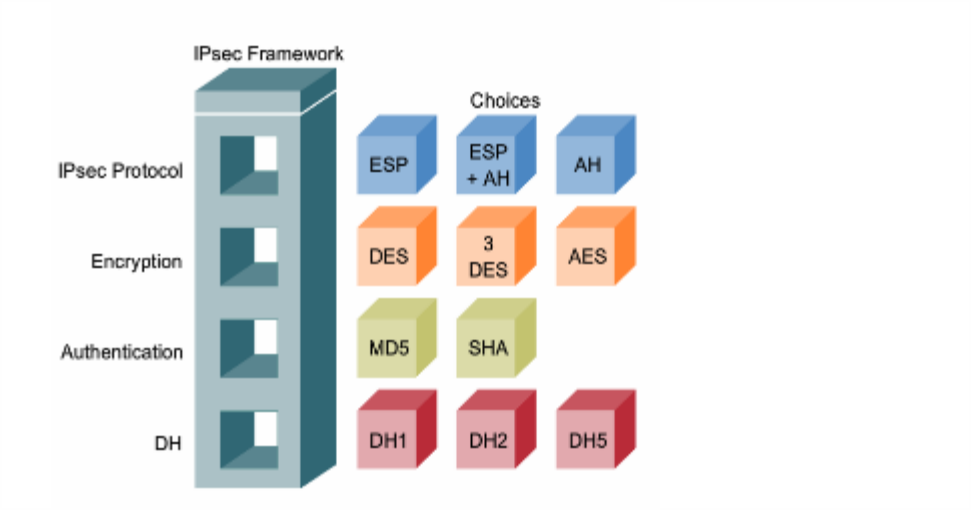
- UDP port 500 for Internet Key Exchange (IKE) negotiation traffic
- UDP port 4500 for IPSec Encapsulating Security Protocol (ESP) traffic.

These protocols must not be blocked by any firewalls or the ISP networks between the two IPSec security devices attempting to establish the tunnel.

NAT Traversal (NAT-T) NAT Traversal (NAT-T) is a VPN option used on many IPSec security devices. It is typically enabled by default. With this option, a NAT discovery process runs after the IKE initiation request to determine if there are any NAT devices in the tunnel path. If a NAT device is detected in the tunnel path, the IPSec security devices will use UDP encapsulated IPSec packets for the VPN tunnel. NAT discovery messages are displayed in the logs, but typically only in the IKE Responder log with Aggressive Mode.

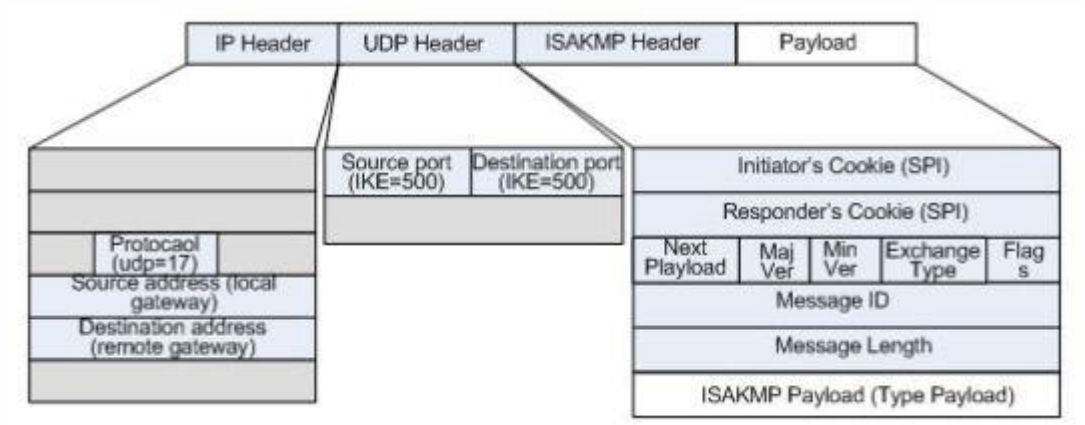
IPSEC:

IPSEC Framework Diagram:



IPSEC Packet Structure:

ISAKMP packet encapsulation and packet headers :



IP packet header

- SRC (Source IP Address): local IP address of the initiated IKE negotiation; may be that of a physical/logical interface and maybe be command configured.
- DST (Destination IP Address): peer IP address of the initiated IKE negotiation; command configured.

UDP packet header

IKE protocol port 500 initiates negotiation and responds to negotiation. When both the host and sub-hosts have fixed IP addresses, this port will never change in the negotiation process. When either the host or the sub-host s have an NAT device (NAT traversal scenario), the IKE protocol will use a special process which we will discuss later on.

ISAKMP packet header

− Initiator's cookie (SPI) and responder's cookie (SPI): the SPI serves as a cookie for both IKEv1 and IKEv2, a unique IKE SA identifier.

− Version: the IKE version number. Many things have changed for the better since the launch of IKEs. To differentiate, older IKEs are known as IKEv1 while updated IKEs are known as IKEv2.

− Exchange Type: the IKE defined exchange type. Exchange types define the exchange sequence that ISAKMP messages must follow. Later, we will discuss the IKEv1 main mode, aggressive mode, and fast mode. When discussing IKEv2, we'll mention initial exchanges and child SA exchanges. All of these are different IKE defined exchange types. Indicates the type of exchange being used. This dictates the message and payload orderings in the ISAKMP exchanges.

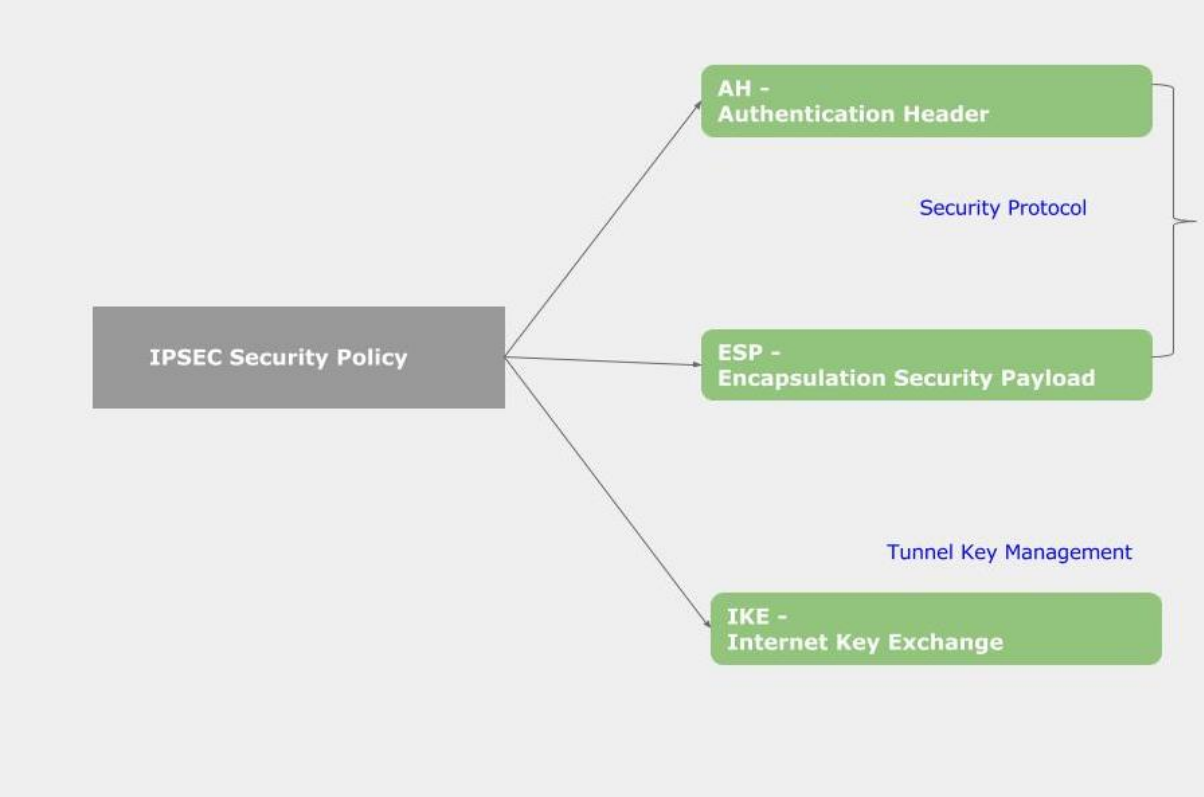
− Next Payload: The next payload type identifies the message. A single ISAKMP packet may be loaded with multiple payloads. This field provides "link" capabilities within the payload. If the current payload is the message's final payload, this field will be 0. Indicates the type of the first payload in the message.

− Message ID. 4 bytes. A unique value used to identify the protocol state during Phase 2 negotiations. It is randomly generated by the initiator of the Phase 2 negotiation.

− Length. 4 bytes.The total length of the ISAKMP header and the encapsulated payloads in bytes.

− ISAKMP Payload (Type Payload): A type of payload carried in an ISAKMP packet that is used as a "parameters packet" for negotiating IKE and IPSec SAs. There are many different types of payloads, and each different payload may carry different "parameter packets". The specific usage of different payloads will be discussed together with the packet capturing process.

IPSEC Architecture:



IPSEC Mode of operation

IPsec can be run in either tunnel mode or transport mode. Tunnel mode is Default.

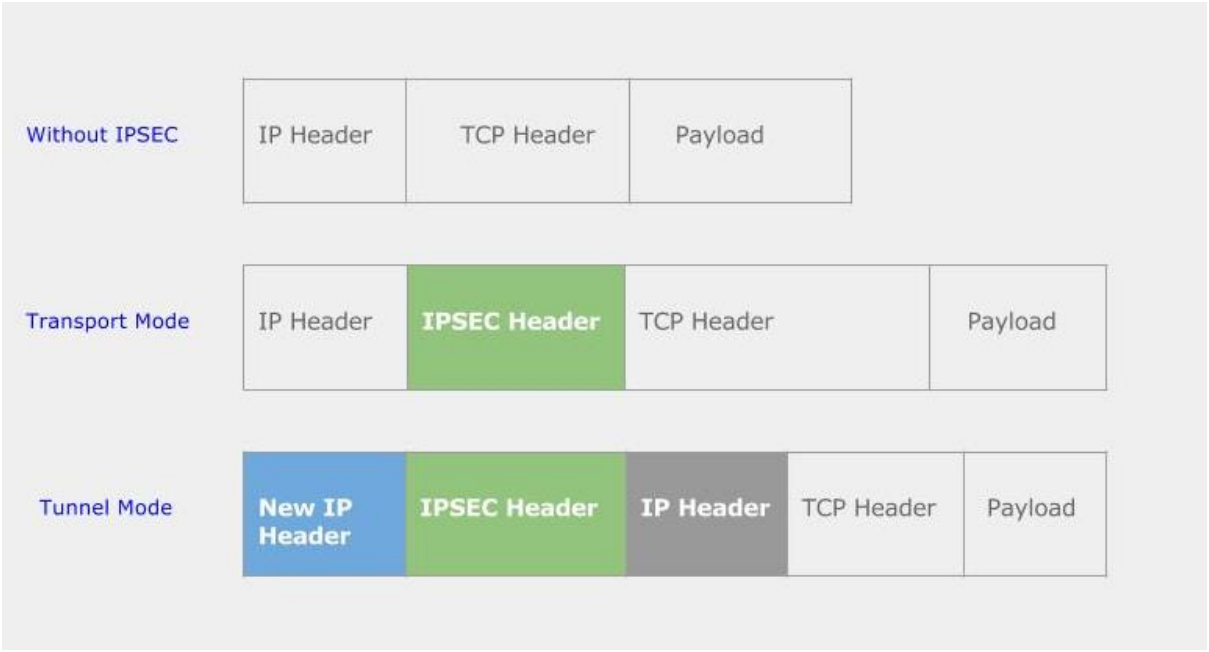
Transport Mode	Tunnel Mode
1. Here, IP header of original data is not encrypted Payload and ESP trailer is encrypted	1. Here, IP header of original data is encrypted
2. IPSEC Header is inserted into IP packet New packet is not created	2. Entire IP packet is encrypted and New packet is created with larger IP packet size
3. Widely used in Client to Site VPN scenarios	3. Widely used site to site VPN scenarios
4. NAT traversal is not supported	4. NAT traversal is supported
5. MSS is higher As additional headers are not required	5. MSS is less As additional headers are required

Where Transport Mode is used ?

As we know, In transport mode, IP header of original packet is not encrypted

So, first some Tunneling protocol like L2TP or GRE is used to encapsulate IP data packet

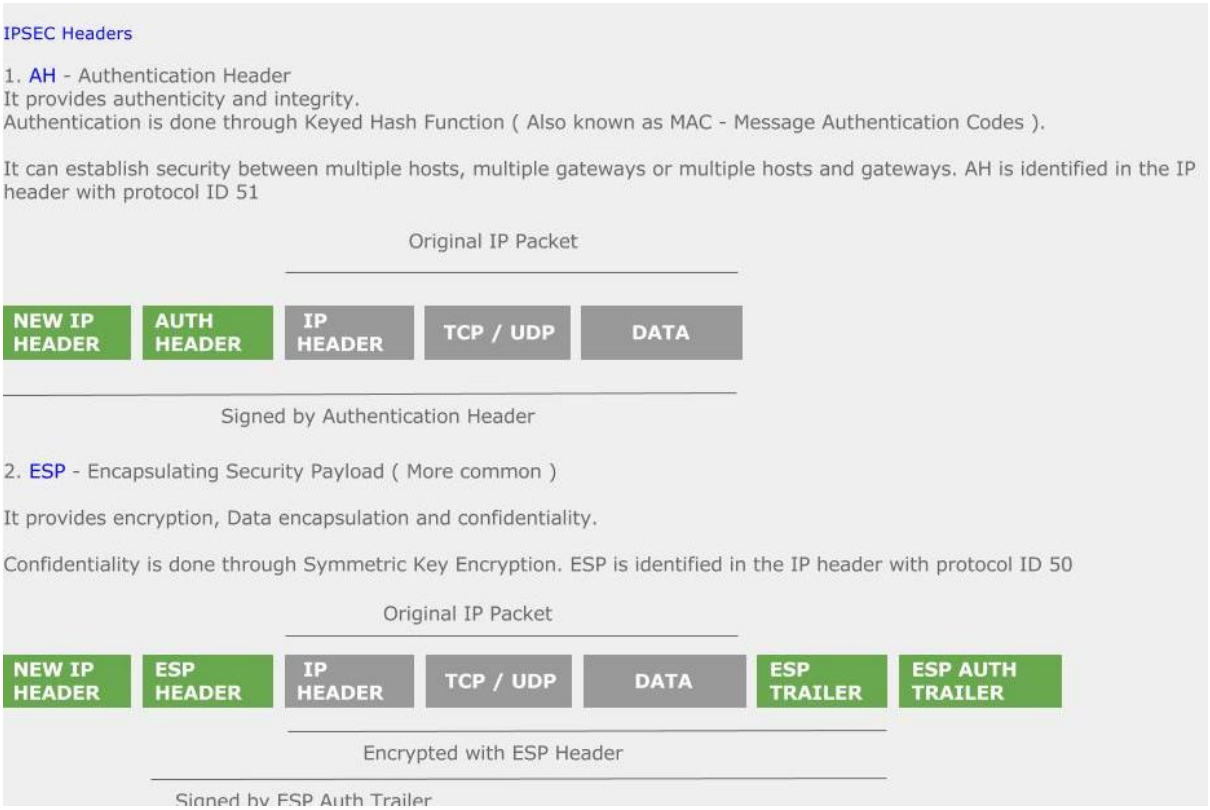
Then, IPSEC is used to protect the Tunneling protocol like L2TP or GR



IPSEC Headers:

1.AH: Authentication Header

2.ESP: Encapsulating Security Payload



IPSEC Operation:

IPSEC Operation

Multiple tunnels are created depending on IPSEC Operation Mode - Like Transport or Tunnel Mode

During the transfer of Data, additional headers are added to the packet

When passing through Gateway, every time Header is added

In Header, SPI (Security Parameter Index) is included

SPI tells the algorithm and keys used by the last node to view the packet

Payload is also protected, any change or error will simply lead to dropping of packet

These headers are added at entrance of tunnel and removed while exiting tunnel

SA (Security Association) uses the SPI number present in the Header

In SA, destination IP address is also included to indicate the end point if it is user, Router or Firewall

SAD (Security Association Database) is used to store all SA's

Security Policy is associated with SAD, which tells what should be done with packet

Example Action

Dropping packet altogether

Dropping SA only

Substituting different SA

IPSec configuration:-

By now we have a step-by-step process for IPSec configuration that we can use:

A.Phase 1

B.Phase 2

Phase 1

Step 1. Configure ISAKMP using pre-shared authentication, MD5 hashing, DH group , and a PSK of "cisco" on both HUB and Spoke.

Step 2. Configure the ISAKMP key and identify the peer.

Phase 2

Step 3. Configure the IPSec transform set to use DES for encryption and MD5 for hashing:

Step 4. Define interesting traffic

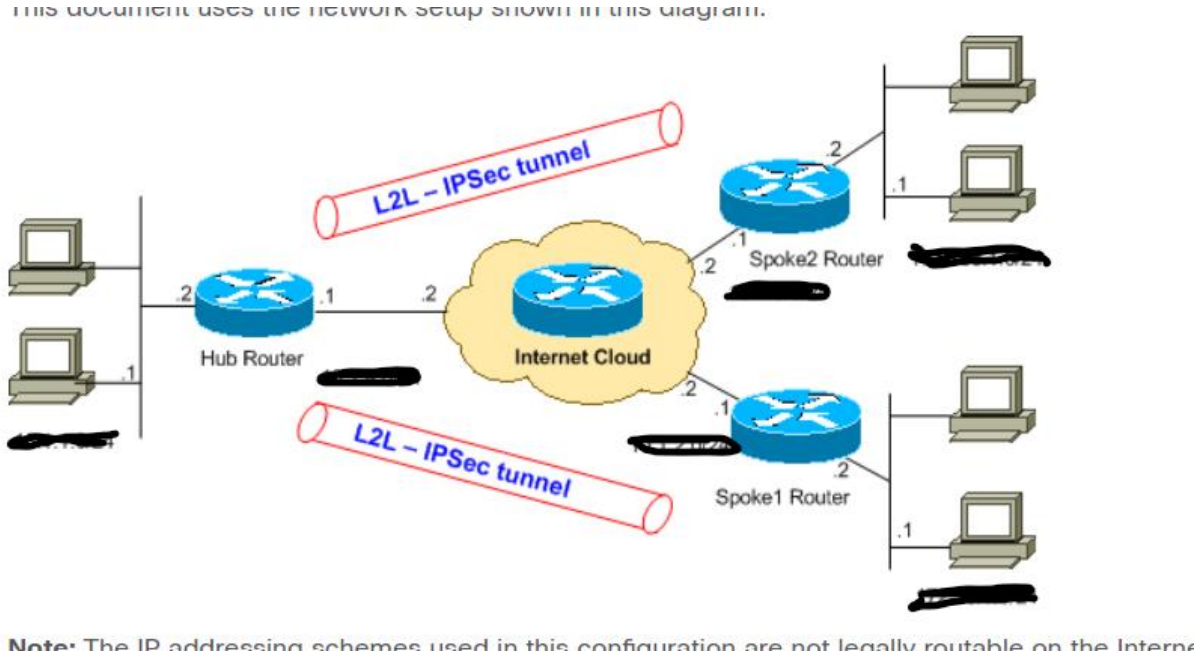
Step 5. Configure a crypto map and reference the peer, the crypto ACL, and the transform set configured in the previous steps.

Apply the crypto map to the outside interface:

TROUBLESHOOTING:

In this part I will be discussing the following problem scenarios…

Topology Like below……..



- **IKE SA not established**
- **IPSEC SA's not established**
- **MTU/Fragmentation Issues**

Problem Scenario 1: No IKE SAs

Issue: If we are unable to establish IPSEC tunnel from Branch location to Hub location
Before Your VPN Can work, You need to check connectivity between the WANs on each peer Router.
Chek the WAN to WAN Connectivity for Routing. Ping the Branch (using HUB's IKE endpoint)

HUB# ping ip 40.10.1.1 source 30.3.1.1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 40.10.1.1, timeout is 2 seconds:
Packet sent with a source address of 30.3.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms

Check for IKE SA

HUB# sh crypto isakmp sa

IPv4 Crypto ISAKMP SA

dst	src	state	conn-id	slot	status
30.3.1.1	40.10.1.1	MM_NO_STATE	0	0	ACTIVE (deleted)

MM_NO_STATE Means:-

State	Explanation
MM_NO_STATE	The ISAKMP SA has been created, but nothing else has happened yet. It is "larval" at this stage.

For an tunnel to be perfectly up and passing traffic like it is supposed to, you should see a status and "**QM_IDLE**" on a router.

IPv6 Crypto ISAKMP SA

- There are several commands that may be used for debugging IPsec connections and we will focus on the most common:
  - `debug crypto isakmp`
  - `debug crypto ipsec`

Use IKE Debugs to troubleshoot [debug crypto isakmp]

Problem Scenario 1a:
No IKE SAs
ISAKMP:(0):Checking ISAKMP transform 1 against priority 10 policy
ISAKMP: encryption 3DES-CBC
ISAKMP: hash SHA
ISAKMP: default group 2
ISAKMP: auth pre-share
ISAKMP: life type in seconds
ISAKMP: life duration (basic) of 7200
ISAKMP:(0):Encryption algorithm offered does not match policy!
ISAKMP:(0):atts are not acceptable. Next payload is 0
ISAKMP:(0):Checking ISAKMP transform 1 against priority 65535 policy
ISAKMP: encryption 3DES-CBC
ISAKMP: hash SHA
ISAKMP: default group 2
ISAKMP: auth pre-share
ISAKMP: life type in seconds
ISAKMP: life duration (basic) of 7200

ISAKMP:(0):Encryption algorithm offered does not match policy!ISAKMP:(0):atts are not acceptable. Next payload is 0
ISAKMP:(0):no offers accepted!ISAKMP:(0): phase 1 SA policy not acceptable! (local 30.3.1.1 remote 40.10.1.1)

Check the IKE Policies

Check the ISAKMP policies that are configured on both the ends of the tunnel to check if the parameters are matched. By ISAKMP policies, I am referring to the parameters that have been configured after issuing the command.

And also it gives:

*Provides The Phase 1 Policy Configuration and is separated by priority number
* Includes a Default Settings

HUB# sh crypto isakmp policy

Global IKE policy
Protection suite of priority 10
encryption algorithm: AES - Advanced Encryption Standard (128 bit keys).
hash algorithm: Secure Hash Standard
authentication method: Pre-Shared Key
Diffie-Hellman group: #2 (1024 bit)
lifetime: 7200 seconds, no volume limit
Default protection suite
encryption algorithm: DES - Data Encryption Standard (56 bit keys).

SPOKE

```
crypto isakmp policy 10
 encr 3des
 authentication pre-share
 group 2
 lifetime 7200
```

HUB

```
crypto isakmp policy 10
 encr aes
 authentication pre-share
 group 2
 lifetime 7200
```

Solution: So once we change the encryption algorithm at spoke side to AES, phase 1 will come up.

Problem Scenario 1b:

No IKE SAs

ISAKMP:(1017): sending packet to 40.10.1.1 my_port 500 peer_port 500 (R) MM_KEY_EXCH

ISAKMP:(1017):Input = IKE_MESG_INTERNAL, IKE_PROCESS_COMPLETE
ISAKMP:(1017):Old State = IKE_R_MM3 New State = IKE_R_MM4
ISAKMP (0:1017): received packet from 40.10.1.1 dport 500 sport 500 Global (R) MM_KEY_EXCH
ISAKMP: reserved not zero on ID payload!
%CRYPTO-4-IKMP_BAD_MESSAGE: IKE message from 40.10.1.1 failed its sanity check or is malformed

HUB

```
crypto keyring SPOKES
    pre-shared-key address 40.10.1.1
  key cisco
```

SPOKE

```
crypto keyring HUB
    pre-shared-key address 30.3.1.1
  key cisco123
```

Solution: It means we have a mismatch in pre-shared key, on correcting it our IKE SA should come up.

HUB# sh cry isakmp sa
This command shows the Internet Security Association Management Protocol (ISAKMP) security associations (SAs) built between peers.

IPv4 Crypto ISAKMP SA

dst	src	state	conn-id	slot	status
30.3.1.1	40.10.1.1	QM_IDLE	1019	0	ACTIVE

=== ==== ========== ========== ================== ==========

Problem Scenario 2: No IPSEC SAs:-

Issue:
If you notice that there is no traffic is being received through the IPSEC tunnel
IKE SAs exist, but no IPSec SAs

Check for IPSEC SA (look for inbound and outbound SPI's)

HUB# sh crypto ipsec sa peer 40.10.1.1

Shows the settings, number of encaps and decaps, local and remote proxy identities, and Security Parameter Indexes (SPIs) (inbound and outbound) used by current Security Associations (SAs)

```
interface: GigabitEthernet0/1
        Crypto map tag: CMAP, local addr 30.3.1.1

        protected vrf: (none)
        local  ident (addr/mask/prot/port): (3.1.1.0/255.255.255.0/0/0)
        remote ident (addr/mask/prot/port): (4.1.1.0/255.255.255.0/0/0)
        current_peer 40.10.1.1 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.: 30.3.1.1, remote crypto endpt.: 40.10.1.1
        path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0/1
        current outbound spi: 0x0(0)

    inbound esp sas:
        inbound ah sas:

        outbound esp sas:
        outbound ah sas:

HUB#
Use IPSec Debugs to troubleshoot [ debug crypto ipsec ]
```

Problem Scenario 2a: No IPSec SAs

```
ISAKMP (0:1022): received packet from 40.10.1.1 dport 500 sport 500 Global (R)
QM_IDLE
ISAKMP:(1022): processing SA payload. message ID = -549695704
ISAKMP:(1022):Checking IPSec proposal 1
ISAKMP: transform 1, ESP_3DES
ISAKMP:   attributes in transform:
ISAKMP:       encaps is 1 (Tunnel)
ISAKMP:       SA life type in seconds
ISAKMP:       SA life duration (basic) of 1800
ISAKMP:       SA life type in kilobytes
ISAKMP:       SA life duration (VPI) of  0x0 0x46 0x50 0x0
ISAKMP:       authenticator is HMAC-SHA
ISAKMP:(1022):atts are acceptable.
IPSEC(validate_proposal_request): proposal part #1,
  (key eng. msg.) INBOUND local= 30.3.1.1, remote= 40.10.1.1,
        local_proxy= 3.1.1.0/255.255.255.0/0/0 (type=4),
        remote_proxy= 4.1.1.0/255.255.255.0/0/0 (type=4),
        protocol= ESP, transform= NONE  (Tunnel),
        lifedur= 0s and 0kb,
        spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x0
Crypto mapdb : proxy_match
        src addr        : 3.1.1.0
        dst addr        : 4.1.1.0
        protocol        : 0
```

src port : 0
dst port : 0
IPSEC(ipsec_process_proposal): transform proposal not supported for identity: {esp-3des esp-sha-hmac }
ISAKMP:(1022): IPSec policy invalidated proposal with error 256ISAKMP:(1022): phase 2 SA policy not acceptable! (local 30.3.1.1 remote 40.10.1.1)

Check the IPSEC Transform Sets

HUB# sh cry ips transform-set
Transform set TS: { esp-aes esp-sha-hmac }
will negotiate = { Tunnel, },

HUB

```
crypto ipsec transform-set esp-aes esp-
   sha-hmac
```

SPOKE

```
crypto ipsec transform-set TS esp-3des
   esp-sha-hmac
```

Solution: On Correcting encryption algorithm in transform-set , tunnel should come up.

Problem Scenario 2b: No IPSEC SAs

Check the Crypto ACLs

HUB# sh access-list SPOKE-10-ACL
Extended IP access list SPOKE10-ACL
10 permit ip 3.1.1.0 0.0.0.255 5.1.1.0 0.0.0.255
HUB#

SPOKE

```
ip access-list extended HUB-ACL
   permit ip 4.1.1.0 0.0.0.255 3.1.1.0
   0.0.0.255
```

HUB

```
ip access-list extended SPOKE10-ACL
    permit ip 3.1.1.0 0.0.0.255 5.1.1.0
    0.0.0.255
```

Solution:- **On Correcting crypto access-list , tunnel should come up.**

Problem Scenario 3: Anti-Replay Issues

Issue:-If you notice that some of the applications are losing intermittent traffic, or that Voice quality through tunnel is bad.

Check if the IPSec SA is showing anti-replay drops

```
HUB# sh cry ips sa peer 40.10.1.1 detail
interface: GigabitEthernet0/1
        Crypto map tag: CMAP, local addr 30.3.1.1

        protected vrf: (none)
        local  ident (addr/mask/prot/port): (3.1.1.0/255.255.255.0/0/0)
        remote ident (addr/mask/prot/port): (4.1.1.0/255.255.255.0/0/0)
        current_peer 40.10.1.1 port 500
        PERMIT, flags={origin_is_acl,}
        #pkts encaps: 2900, #pkts encrypt: 2900, #pkts digest: 2900
        #pkts decaps: 1909, #pkts decrypt: 1909, #pkts verify: 1909
        #pkts compressed: 0, #pkts decompressed: 0
        #pkts not compressed: 0, #pkts compr. failed: 0
        #pkts not decompressed: 0, #pkts decompress failed: 0
        #pkts no sa (send) 0, #pkts invalid sa (rcv) 0
        #pkts encaps failed (send) 0, #pkts decaps failed (rcv) 0
        #pkts invalid prot (recv) 0, #pkts verify failed: 0
        #pkts invalid identity (recv) 0, #pkts invalid len (rcv) 0
        #pkts replay rollover (send): 0, #pkts replay rollover (rcv) 0
   ##pkts replay failed (rcv): 1000  #pkts internal err (send): 0, #pkts internal err (recv)
0

        local crypto endpt.: 30.3.1.1, remote crypto endpt.: 40.10.1.1
        path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0/1
        current outbound spi: 0xC37422AA(3279168170)

        inbound esp sas:
        spi: 0x135E76B1(324957873)
        transform: esp-3des esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 41, flow_id: SW:41, crypto map: CMAP
        sa timing: remaining key lifetime (k/sec): (4419198/860)
        IV size: 8 bytes
      replay detection support: Y
        Status: ACTIVE
```

Default IPSec Anti-Replay window is 64
Packets received outside the window are dropped
Re-ordering of packets could happen due to QoS on the encrypting router (Spoke) or in the Transit Network
In current Cisco IOS versions, the Anti-Replay window can be increased up to 1024, or diabled altogether

crypto ipsec security-association window-size

crypto ipsec security-association replay disable

Not recommended to disable anti-replay; first try to fix the QoS issue in the network or encrypting router; give better QoS to Voice traffic, or use crypto LLQ; then try to increase the anti-replay window size.

IPSec Troubleshooting: Problem Scenarios Part 2

In this part 2 will be discussing the following problem scenarios----

- **Routing Issues (Reverse Route Injection)**
- **DPD**
- **Anti-Replay**

Problem Scenario 1:

Routing Issues

Issue:User complains there is no traffic received through the IPSec tunnel. On further checking you find that IKE and IPSec SAs exist, but no end-end traffic; spoke shows its encrypting traffic however no decrpyt.

Check for IPSec SA on Hub Site (look for inbound and outbound SPIs, encr/decr counts)

HUB# sh crypto session remote 40.10.1.1 detail
Crypto session current status

Code: C - IKE Configuration mode, D - Dead Peer Detection
K - Keepalives, N - NAT-traversal, X - IKE Extended Authentication

Interface: GigabitEthernet0/1
Profile: SPOKE10-PROF
Uptime: 00:01:49
Session status: UP-ACTIVE
Peer: 40.10.1.1 port 500 fvrf: (none) ivrf: (none)
Phase1_id: 40.10.1.1
Desc: (none)
IKE SA: local 30.3.1.1/500 remote 40.10.1.1/500 Active
Capabilities:D connid:1029 lifetime:01:58:10
IPSEC FLOW: permit ip 3.1.1.0/255.255.255.0 4.1.1.0/255.255.255.0
Active SAs: 2, origin: crypto map

```
        Inbound:  #pkts dec'ed 9949 drop 60 life (KB/Sec) 4483560/1690
        Outbound: #pkts enc'ed 0 drop 0 life (KB/Sec) 4485046/1690
HUB# sh crypto ipsec sa peer 40.10.1.1

interface: GigabitEthernet0/1
        Crypto map tag: CMAP, local addr 30.3.1.1

        protected vrf: (none)
        local  ident (addr/mask/prot/port): (3.1.1.0/255.255.255.0/0/0)
        remote ident (addr/mask/prot/port): (4.1.1.0/255.255.255.0/0/0)
        current_peer 40.10.1.1 port 500
        PERMIT, flags={origin_is_acl,}
    #pkts encaps: 0, #pkts encrypt: 0, #pkts digest: 0
    #pkts decaps: 9949, #pkts decrypt: 9949, #pkts verify: 9949
        #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 60

        local crypto endpt.: 30.3.1.1, remote crypto endpt.: 40.10.1.1
        path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0/1
        current outbound spi: 0xF6278D63(4129787235)

        inbound esp sas:
        spi: 0x16C58DD4(382045652)
        transform: esp-3des esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 27, flow_id: SW:27, crypto map: CMAP
        sa timing: remaining key lifetime (k/sec): (4483560/1659)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE

        inbound ah sas:

        inbound pcp sas:

        outbound esp sas:
        spi: 0xF6278D63(4129787235)
        transform: esp-3des esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 28, flow_id: SW:28, crypto map: CMAP
        sa timing: remaining key lifetime (k/sec): (4485046/1657)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE

        outbound ah sas:

        outbound pcp sas:
```

Check the Routes (for the Spoke protected networks)

```
HUB# sh ip route 4.1.1.1
```

% Network not in table
HUB#

HUB# sh ip cef 4.1.1.1
0.0.0.0/0, version 80, epoch 0, attached, default route handler
0 packets, 0 bytes
  via 0.0.0.0, 0 dependencies
   valid no route adjacency

HUB#
Check the Crypto Map for Reverse-Route Injection. This is needed for the Hub to inject a route for the Spoke protected subnets into its local routing table. The route is created when the IPSec SA is established. Since 12.4T, for this route to be created (based on the Crypto ACL) before the IPSec SA is established (so that the router can initiate the tunnel), we need the “reverse-route static” configuration.
In the VRF-Aware IPSec scenario, it is better to use the “reverse-route remote-peer <next-hop-gateway>” configuration under the crypto map.
Old Crytpo Map was----
crypto map CMAP 10 ipsec-isakmp
 set peer 40.10.1.1
 set transform-set TS
 set isakmp-profile SPOKE10-PROF
 match address SPOKE10-ACL
Lets add reverse route---
crypto map CMAP 10 ipsec-isakmp
 set peer 40.10.1.1
 set transform-set TS
 set isakmp-profile SPOKE10-PROF
 match address SPOKE10-ACL
reverse-route <static>
IPSEC(crypto_ipsec_sa_find_ident_head): reconnecting with the same proxies and peer 40.10.1.1
IPSEC(rte_mgr): VPN Route Event create SA based on crypto ACL in real time for 40.10.1.1IPSEC(rte_mgr): VPN Route Refcount 1 GigabitEthernet0/1
IPSEC(rte_mgr): VPN Route Added 4.1.1.0 255.255.255.0 via 0.0.0.0 in IP DEFAULT TABLE with tag 0 distance 1

HUB# sh ip route 4.1.1.1
Routing entry for 4.1.1.0/24
  Known via "static", distance 1, metric 0
 Redistributing via ospf 1
  Advertised by ospf 1 subnets
  Routing Descriptor Blocks:
  * 40.10.1.1
        Route metric is 0, traffic share count is 1

HUB# sh ip cef 4.1.1.1
4.1.1.0/24, version 83, epoch 0, cached adjacency 30.3.1.2
0 packets, 0 bytes
  via 40.10.1.1, 0 dependencies, recursive
        next hop 30.3.1.2, GigabitEthernet0/1 via 40.0.0.0/8
        valid cached adjacency

Complete Crypto Map

```
HUB# show crypto map
Crypto Map "CMAP" 10 ipsec-isakmp
        Peer = 40.10.1.1
        ISAKMP Profile: SPOKE10-PROF
        Extended IP access list SPOKE10-ACL
        access-list SPOKE10-ACL permit ip 3.1.1.0 0.0.0.255 4.1.1.0 0.0.0.255
      Current peer: 40.10.1.1
        Security association lifetime: 4608000 kilobytes/1800 seconds
        PFS (Y/N): N
        Transform sets={
                TS,
        }
        Reverse Route Injection Enabled
        Interfaces using crypto map CMAP:
                GigabitEthernet0/1
==== =========== ============== ============== ============
```

Problem Scenario 2:

DPD(Dead Peer Detection): is a method that allows detection of unreachable Internet Key Exchange (IKE) peers.

Issue:This is a scenario where HUB keeps sending encrypted traffic, but it is not receiving any encrypted traffic from Spoke. IKE and IPSec SAs are up.

Please perform the following steps------

Check if the Spoke is reachable (ping tunnel endpoint address)

```
HUB# ping 40.10.10.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 40.10.10.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
No replies.
```

Check if Dead peer Detection is turned on

```
HUB# sh cry isa sa detail

Codes: C - IKE configuration mode, D - Dead Peer Detection
        K - Keepalives, N - NAT-traversal
        X - IKE Extended Authentication
        psk - Preshared key, rsig - RSA signature
        renc - RSA encryption

IPv4 Crypto ISAKMP SA

C-id    Local           Remote          I-VRF  Status        Encr  Hash  Auth  DH
        Lifetime        Cap.
1035  30.3.1.1          40.10.1.1              ACTIVE        3des  sha     psk     2
        01:59:45
        Engine-id:Conn-id =  SW:35
```

So Under Cap, its not listing anything, hence it is disabled.

Check the IPSec SA

```
HUB#sh cry ips sa

interface: GigabitEthernet0/1
        Crypto map tag: CMAP, local addr 30.3.1.1

        protected vrf: (none)
        local  ident (addr/mask/prot/port): (3.1.1.0/255.255.255.0/0/0)
        remote ident (addr/mask/prot/port): (4.1.1.0/255.255.255.0/0/0)
        current_peer 40.10.1.1 port 500
        PERMIT, flags={origin_is_acl,}
        #pkts encaps: 100, #pkts encrypt: 100, #pkts digest: 100
        #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

        local crypto endpt.: 30.3.1.1, remote crypto endpt.: 40.10.1.1
        path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0/1
        current outbound spi: 0x8BDBBA86(2346433158)

        inbound esp sas:
        spi: 0x67C89AAD(1741200045)
```

We notice very few decaps, lets go ahead and configure DPD.

Configure DPD

```
        crypto isakmp keepalive 60 10
        crypto isakmp keepalive 60 periodic
```

If DPD had been configured earlier, then you would have seen following-----

```
HUB# sh cry isakmp peer de
Peer: 40.10.1.1 Port: 500 Local: 30.3.1.1
  Phase1 id: 40.10.1.1
   flags:
  NAS Port: 0 (Normal) DPD information, struct 0x6727E0E8:
   Last_received: 237, dpd threshold (elapsed) 0
   my_last_seq_num: 0x5B72ECCC, peers_last_seq_num: 0x0
   sent_and_waiting: TRUE
  IKE SAs: 1 IPSec SA bundles: 1
   last_locker: 0x62FE32FC, last_last_locker: 0x0
   last_unlocker: 0x0, last_last_unlocker: 0x0

HUB# sh cry isa sa detail

Codes: C - IKE configuration mode, D - Dead Peer Detection
        K - Keepalives, N - NAT-traversal
        X - IKE Extended Authentication
        psk - Preshared key, rsig - RSA signature
        renc - RSA encryption
```

IPv4 Crypto ISAKMP SA
C-id Local Remote I-VRF Status Encr Hash Auth DH Lifetime Cap.

1037 30.3.1.1 40.10.1.1 ACTIVE 3des sha psk 2 01:59:43 D
Engine-id:Conn-id = SW:37
PS: It shows D under Cap now.
ISAKMP:(1036):DPD incrementing error counter (4/5)
ISAKMP: set new node 1992211651 to QM_IDLE
ISAKMP:(1036):Sending NOTIFY DPD/R_U_THERE protocol 1
spi 1718567840, message ID = 1992211651
ISAKMP:(1036): seq. no 0x5B72ECCD
ISAKMP:(1036): sending packet to 40.10.1.1 my_port 500 peer_port 500 (R) QM_IDLE
ISAKMP:(1036):Sending an IKE IPv4 Packet.
ISAKMP:(1036):purging node 1992211651
ISAKMP:(1036):Input = IKE_MESG_FROM_TIMER, IKE_TIMER_PEERS_ALIVE
ISAKMP:(1036):Old State = IKE_P1_COMPLETE New State = IKE_P1_COMPLETE

ISAKMP:(1036):DPD incrementing error counter (5/5)
ISAKMP:(1036):peer 40.10.1.1 not responding!
ISAKMP:(1036):peer does not do paranoid keepalives.

ISAKMP:(1036):deleting SA reason "P1 errcounter exceeded (PEERS_ALIVE_TIMER)" state (R) QM_IDLE (peer 40.10.1.1)

It is always better to use DPD instead of Periodic Keepalives. DPD works well in conjunction with IPSec HA – geographically distributed peers (multiple 'set peer' under crypto map), or HSRP adjacent peers (peer to VIP address).

Problem Scenario 3:

Anti-Replay Issues

Issue:Users complain that application is losing intermittent traffic, or that Voice quality through tunnel is bad. Check if the IPSec SA is showing anti-replay drops

HUB# sh cry ips sa peer 40.10.1.1 detail

interface: GigabitEthernet0/1
Crypto map tag: CMAP, local addr 30.3.1.1
protected vrf: (none)
local ident (addr/mask/prot/port): (3.1.1.0/255.255.255.0/0/0)
remote ident (addr/mask/prot/port): (4.1.1.0/255.255.255.0/0/0)
current_peer 40.10.1.1 port 500
PERMIT, flags={origin_is_acl,}
#pkts encaps: 2900, #pkts encrypt: 2900, #pkts digest: 2900
#pkts decaps: 1909, #pkts decrypt: 1909, #pkts verify: 1909
#pkts compressed: 0, #pkts decompressed: 0
#pkts not compressed: 0, #pkts compr. failed: 0
#pkts not decompressed: 0, #pkts decompress failed: 0
#pkts no sa (send) 0, #pkts invalid sa (rcv) 0
#pkts encaps failed (send) 0, #pkts decaps failed (rcv) 0
#pkts invalid prot (recv) 0, #pkts verify failed: 0

```
        #pkts invalid identity (recv) 0, #pkts invalid len (rcv) 0
        #pkts replay rollover (send): 0, #pkts replay rollover (rcv) 0
    ##pkts replay failed (rcv): 1000
        #pkts internal err (send): 0, #pkts internal err (recv) 0

        local crypto endpt.: 30.3.1.1, remote crypto endpt.: 40.10.1.1
        path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0/1
        current outbound spi: 0xC37422AA(3279168170)

        inbound esp sas:
        spi: 0x135E76B1(324957873)
        transform: esp-3des esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 41, flow_id: SW:41, crypto map: CMAP
        sa timing: remaining key lifetime (k/sec): (4419198/860)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE
```

So we can see there is a good number of packets replay failed in above show commands.

By Default IPSec Anti-Replay window size is 64. Hence, Packets received outside the window will be dropped. Normally Re-ordering of packets could happen due to QoS on the encrypting router (Spoke) or in the Transit Network.
In current Cisco IOS versions, the Anti-Replay window can be increased up to 1024, or diabled altogether

```
        crypto ipsec security-association window-size < Size>

        crypto ipsec security-association replay disable
```

It is not recommended to disable anti-replay. Hence first try to fix the QoS issue in the network or encrypting router; give better QoS to Voice traffic, or use crypto LLQ; then try to increase the anti-replay window size by above mentioned command.

========== ============== ================ =================

Summary

- **show crypto isakmp sa** and **show crypto ipsec sa** will show you the Phase 1 and Phase 2 tunnel information respectively
- **show crypto connections active** is a quick way to view phase one and phase 2 tunnels and obtain information on packets encrypted and decrypted
- **show ip nat translations** provides information about current translation entries and is a quick way to troubleshoot IPsec and NAT coexistence
- **debug crypto isakmp** and **debug crypto ipsec** provide detailed information about IPsec tunnel negotiation and assist with identifying configuration issues

- To show IPSec SA information:
  - **show crypto ipsec sa [address | detail | interface | map | per | vrf]**
- To show IKE and IPSec information together :
  - **show crypto session [fvrf | group | ivrf] username | detail]**
  - **show crypto engine connection active**

Cisco IOS IPSec Debugging

- These are the current IKE/IPSec debugs available; the highlighted ones are the most useful typically
- Make sure to use Crypto Conditional Debugs when trying to troubleshoot production routers

debug crypto isakmp

debug crypto isakmp error

debug crypto isakmp ha

debug crypto ipsec

debug crypto ipsec error

debug crypto routing

debug crypto ha

debug crypto engine error

debug crypto engine packet

Clearing VPN Tunnel

- To clear IKE Phase (Phase 1)
  - **clear crypto isakmp sa**
- To clear IPSEC Phase (Phase2)
  - **clear crypto ipsec sa**