Understanding IKEv2 (IKEv1 vs IKEv2)

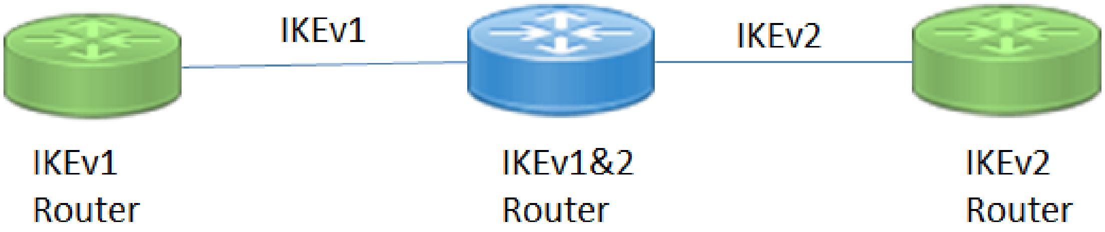
- Internet Key Exchange Version 2 (IKEv2), a next-generation key management protocol based on RFC 4306 and updated in RFC 5996, is a replacement of the IKEv1 Protocol. IKEv2 is used for performing mutual authentication and establishing and maintaining security associations (SAs).

- FlexVPN is Cisco's implementation of the IKEv2 standard featuring a unified paradigm and CLI that combines site to site, remote access, hub and spoke topologies and partial meshes (spoke to spoke direct). FlexVPN offers a simple but modular framework that extensively uses the tunnel interface paradigm while remaining compatible with legacy VPN implementations using crypto maps.

- Crypto maps are considered a legacy configuration construct. It is recommended that you migrate existing crypto map based setups to use tunnel protection and virtual interfaces.

I will explain IKEv2 in the following points with compression with IKEv1 .

1. IKEv2 is not a new IKE version bringing enhancements to IKEv1 but complete new design protocol doing the same objective of IKEv1 which protect user traffic using IPSec.
2. IKEv2 is not backward compatible with IKEv1, still you can have one router running both IKEv1 & IKEv2 one for each point to point link , but any two VPN endpoints both must be configured for IKEv1 or IKEv2.



3. IKEv2 does not consume as much bandwidth as IKEv1.
4. IKEv2 supports EAP authentication while IKEv1 doesn’t.
   IKEv2 support three authentication methods : PSK , PKI (RSA-Sig) , EAP (initiator only) normally when we say initiator we mean client while responder is the server
5. IKEv2 supports MOBIKE while IKEv1 doesn’t. (MOBIKE allows IKEv2 to be used in mobile platforms like phones and by users with multi-homed setups.)
6. IKEv2 has built-in NAT traversal while IKEv1 use it as optional option.
7. IKEv2 can detect whether a tunnel is still alive while IKEv1 can only do that using Dead Peer Detection”, or DPD. DPD is now standard in IKEv2, but it is disabled on IOS by default. It is configurable under the IKEv2 profile; for DPD to be negotiated, both peers must have it enabled.
8. IKEv2 has reliability with acknowledgment and sequenced while IKEv1 is not , in another meaning IKEv1 informational/notification messages are not acknowledged, whereas in IKEv2 they must be acknowledged.

9. IKEv2 generates only 4 messages at all while IKEv1 phase 1 generates in main mode 6 messages and in aggressive mode generates 3 messages , These four message types are: IKE_SA_INIT, IKE_AUTH, CREATE_CHILD_SA, and Informational.
   Phase 1 from IKEv1, which has two functional modes (Main and Aggressive), is known in IKEv2 as IKE_SA_INIT and has a single functional mode requiring two messages to be exchanged. Within a single policy (known as proposal on IOS and policy on ASA), multiple encryption/integrity/PRF/DH groups can be specified in an OR fashion.
   After IKE_SA_INIT derives the keying material, mutual authentication is performed through IKE_AUTH, which requires two messages to be exchanged.
   Phase 2 from IKEv1 (Quick Mode) is known in IKEv2 as CREATE_CHILD_SA.
   Simply , In IKEv2 there is no Main/Aggressive/Quick Modes
10. IKEv2 has facility to negotiate multiple sets of selectors. Many networks/ranges can be negotiated in one exchange. Hence, number of policy records can be very less when sites have multiple networks. while In IKEv1, each pair of networks need to be defined in one policy record in SPD.
11. FLEX VPN work with IKEV2 not IKEV1 , GET VPN still only supports IKEv1, and the EzVPN server/remote functionality from IKEv1 has been replaced by FlexVPN server/client in IKEv2.
12. certificates can be referenced through URL and hash instead of being sent within packets to avoid fragmentation
13. IKEv2 not process request till it determines the requester which make IKEv2 DoS Resilience
14. IKEv2 gives you option to have two crypto engines one handle IPv4 traffic and one for IPv6
15. IKEv1 requires symmetric authentication (both peers using the same method), whereas IKEv2 allows for asymmetric authentication (for example, one side with RSA-SIG and one wide with PSK, or even different PSK).
16. IKEv1 policies called with IKEv2 it proposals , which allows for IKEv2 ID to be protected at all times.
17. IKEv1 does not have a built-in Anti-DOS function, whereas IKEv2 does through anti-clogging cookies.
18. IKEv1 requires re-authentication for IKE rekeying, whereas IKEv2 does not.
19. In IKEv1 IPsec VPNs, ISAKMP/IKEv1 profiles were optional but recommended, but in IKEv2 IPsec VPNs, IKEv2 profiles are required
20. IKEv2 still as IKEv1 use for IKE & ISAKMP UDP protocol with port# 500 and if there is NAT exists in the tunnel path will sue port#4500

Flex VPN (IKEv2) Components:

1. Proposal
2. Policy
3. Keyring
4. Profile

Proposal is a set of algorithms used to protect IKE_SA_INIT
We can define Multiple algorithms such as encryption/integrity etc ,for the same feature.

> **IKEv2 proposal** defines the integrity/encryption/DH group settings and replaces the IKEv1 policies; IKEv2 proposals use names instead of priority numbers, and within each proposal multiple algorithms of each type can be defined, the order of configuration being important. For example, if **3DES** and **AES128** are configured in this order, **3DES** will be negotiated first with the remote peer.

```
crypto ikev2 proposal PROP1
encryption aes-cbc-128 3des
integrity sha md5
group 5 2
```

Policy defines Proposal and matching criteria
Remember , authentication method is no longer negotiated

> **IKEv2 policy** did not exist in IKEv1, and its scope is that you can now restrict which IKEv2 proposals will be negotiated with each VPN peer based on several factors: the local VPN IP address, the FVRF on which the VPN is terminated, and the IP address of the remote VPN peer (not yet implemented).

```
crypto ikev2 policy POL1
proposal PROP1
match fvrf [fvrf_name|any]
match address local IPv4_or_IPv6_address
```

Keyring is a repository of Pre-Shared Keys

```
crypto ikev2 keyring KRING
peer peer_name
hostname name
address IP_or_IPv6_address
identity [address|fqdn|email|key-id] IKEv2_id  < used only in responder
pre-shared-key [local|remote] key_string
```

If you don't specify 'local' or 'remote' keywords for PSK, it will be symmetric
Example of asymmetric Pre-Shared Keys :

```
R1                                   R2
Pre-shared-key local cisco           Pre-shared-key local cisco123
Pre-shared-key remote cisco123       Pre-shared-key remote cisco
```

Profile is a container for all non-negotiable IKEv2 parameters/settings.
Examples :
Authentication method
Keyring/Trustpoint
Authorization options, Lifetime (now NOT negotiated) and more

```
crypto ikev2 profile profile_name
match [options]
authentication {local|remote {rsa-sig|pre-share|ecdsa-sig}
dpd interval retry-interval {on-demand | periodic}
identity local {address | dn | email | fqdn | key-id}
keyring name
ivrf name
nat-keepalive value
pki trustpoint label [sign | verify]
virtual-template number
```

Remember, Profile MUST be always attached to the IPSec Profile

Profile matching options (match) :
- Local IPv4/IPv6 address
- Certificate Map
- FVRF
- IKEv2_ID of the remote peer (IPv4/IPv6 address, e-mail, fqdn, key-id)

Multiple match statements of the same type are logically ORed and multiple match statements of different types are logically ANDed
- Certificate Map and IKEv2_ID are considered to be the same type

```
match vrf cust1
match local address 10.1.1.1
match local address 10.1.2.2
match certificate remote CertMap
```

The result is of above is : „VRF cust1 AND (local ip 10.1.1.1 OR 10.1.2.2) AND peer’s certificate matches CertMap”

So Simply , to run IKEv2 we use the following :

- ○ Proposal (aka policy in IKEv1)
- ○ Keyring (to define the keys will be used)
- ○ Policy
- ○ Profile
- ○ IPsec Transform-set
- ○ IPsec Profile

IKEv2 on the ASA

- No Phase I Proposals; same IKEv2 policy is used instead like in IKEv1
- No Smart Defaults (no default IKEv2 Policy)
- Authentication method is set under Tunnel Group
- Transform Set is now called IPSec Proposal
- Always look for ikev2 keyword for IKEv2 and ipv6 for version 6 IP

There is a nice shortcut on the ASA :
- Configure IKEv1 and convert it to IKEv2 with a single command **migrate {l2l | remote-access {ikev2 | ssl} | overwrite}**

The IKEv2 Smart Defaults

IKEv2 configuration can be simplified by using something called Smart Defaults

Smart Defaults is a group of pre-defined settings for some of IKEv2 Components:
Proposal, Policy, Transform-Set , IPSec Profile and Authorization Policy – all called „default"

To view any of them , use respective show commands along with the „default" keyword such as show crypto ikev2 proposal default
Alternative is to use „show running-config all

The IKEv2 Smart Defaults feature minimizes the FlexVPN configuration by covering most of the use cases. We will practice that later .

Remember Flex VPN smart defaults are :
IKEv2: proposal, policy, profile
IPsec : transform-set , profile

Here is the list of commands that are enabled with the IKEv2 Smart Defaults feature, along with the default values

```
Device# show crypto ikev2 authorization
policy default
IKEv2 Authorization policy: default
route set interface
route accept any tag: 1 distance: 2
```

```
Device# show crypto ikev2 proposal default
IKEv2 proposal: default
Encryption: AES-CBC-256 AES-CBC-192
AES-CBC-128
Integrity: SHA512 SHA384 SHA256 SHA96 MD596
PRF: SHA512 SHA384 SHA256 SHA1 MD5
DH Group: DH_GROUP_1536_MODP
```

```
Device# show crypto ikev2 policy default
IKEv2 policy: default
Match fvrf: any
Match address local: any
Proposal: default
```

```
Device# show crypto ipsec profile default
IPSEC profile default
Security association lifetime: 4608000
kilobytes/3600 seconds
Responder-Only (Y/N): N
PFS (Y/N): N
Transform sets={
default: { esp-aes esp-sha-hmac },
}
```

```
Device# show crypto ipsec transform-set default

Transform set default: { esp-aes esp-sha-hmac
}
will negotiate = { Tunnel, },
```

All or just part of the smart defaults used by IKEv2/IPsec can be disabled so as not to be used by the router in any negotiations (optionally, smart-default settings can also by modified).

For example, to disable the default IPsec transform-set or other smart default :

```
no crypto ikev2 policy default
no crypto ikev2 proposal default
no crypto ipsec profile default
no crypto ipsec transform-set default
```

To re-enable

```
default crypto ipsec transform-set
```

And if we want to modify default

```
crypto ikev2 proposal default
encryption aes-cbc-128
integrity md5

crypto ipsec transform-set default esp-gcm 256
```

Dead Peer Detection (DPD)

Disabled by default but when enable it both peers must have it enabled. When you enable DPD:

- Define the interval for DPD messages if acknowledgement is received (which are informational messages in IKEv2).
- Define the interval for DPD messages if the last DPD message was not acknowledged.
- Define the mode: periodic or on-demand (the same as in IKEv1).
- Use a non-configurable value of 6 retransmits after the last non-acknowledged DPD message.

Example:

```
crypto ikev2 profile R1_TO_R2_P
dpd 30 15 periodic
lifetime 7200
```

In above example we made sure that all SA's should have lifetime of 2 hours and should be deleted within 2 minutes after detecting failure .

IKEv2 Configuration

Finally and simply the following are the configuration we need to use IKEv2:

1-Create crypto ikev2 proposal or use the default

2-Create crypto ikev2 policy or use the default

If we create our own ikev2 proposal & policy we should bind the ikev2 proposal to it

Example::

```
crypto ikev2policy MYPOLICY
proposal MYPROPOSAL
```

3-Create crypto ikev2 keyring

4-Create crypto ikev2 profile

Bind ikev2 keyring to your ikev2 profile

Example:

```
crypto ikev2 profile MYPROFILE
keyring MYRING
```

5-Create crypto ipsec transform-set or use the default

6-Create crypto ipsec profile or use the default

Bind ipsec transform-set (if not using the default) and ikev2 profile to your ipsec profile

example:

```
crypto ipsecprofile MYIPSECPORIFLE
set transform-set MYTS
set crypto ikev2 profile MYPROFILE
```

<u>What is Flex VPN ?</u>

- In IKEv1 we have multiple ways to configuring IPsec VPNs on IOS such as crypto map, DVTI, SVTI, EzVPN, DMVPN, GET VPN.

- Cisco created a new IPsec configuration framework named FlexVPN, which unifies all types of VPNs into a common command and configuration block , FlexVPN will use IKEv2 not IKEv1 .

- IKEv2 as framework tie crypto map (which is no more common used) , Easy VPN , DMVPN , static & dynamic VTI…all together in one set of commands .

- The core of FlexVPN is based on DVTI and as any DVTI we will use virtual-templates , and when IPsec tunnel is successfully negotiated, a virtual-access interface is automatically created that allows per-peer specific configurations to be applied such as QoS, ACL, firewall, using either the local AAA functionality or a remote RADIUS server.

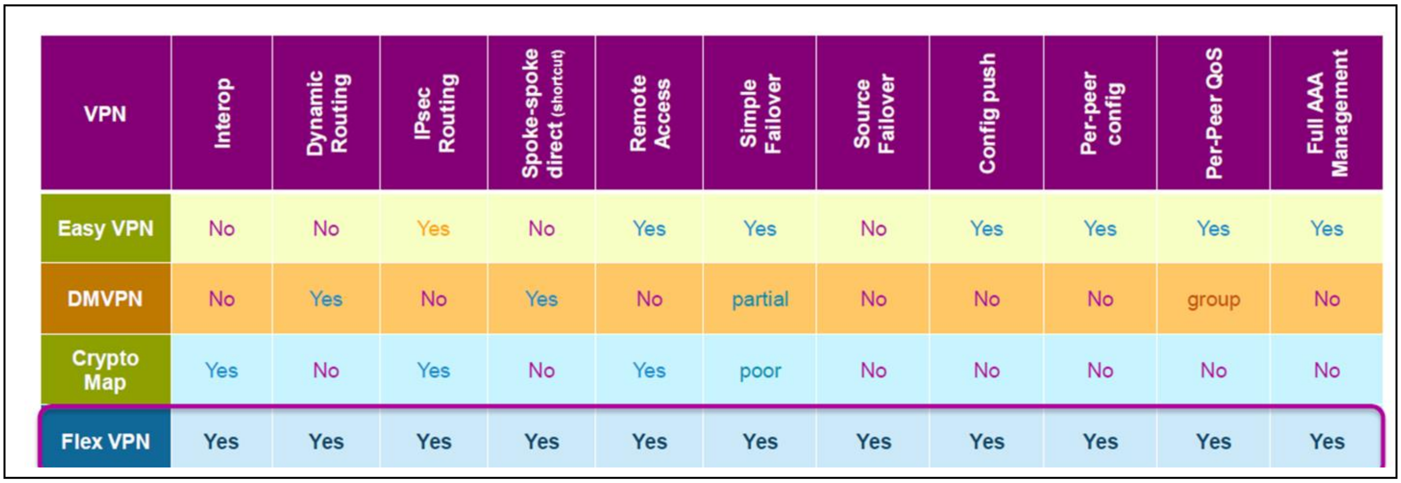
- Flex VPN Server/client used as replacement for EasyVPN Server/Remote

- Same as IKEv1 EasyVPN Server/Remote , in Flex VPN Server/client , the client can be hardware (IOS device) or software (compatible IKEv2 clients such as AnyConnect or the one built in to Windows 7 and Windows 8)

- Also we should remember FlexVPN server uses a regular DVTI setup, while FlexVPN client use a special SVTI setup (crypto map is not supported for FlexVPN server/client).

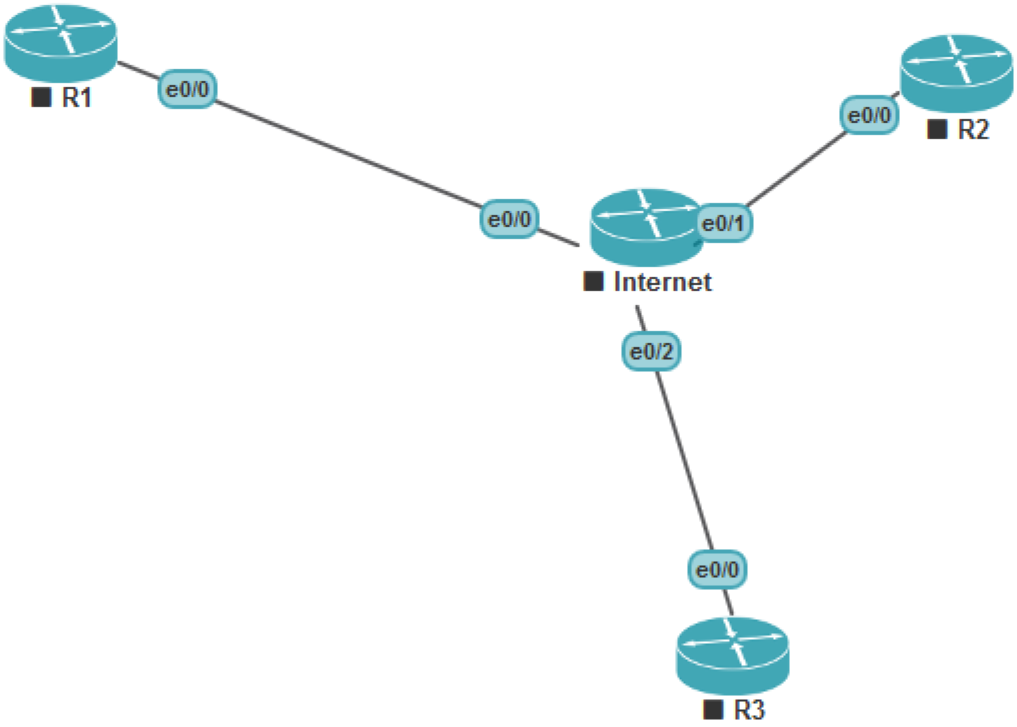
- With FlexVPN, RRI is no longer used. Instead, IKEv2 routing is implemented, which allows each VPN peer to install one or more routes in the routing table of the remote VPN peer; this configuration is done at the IKEv2 authorization policy level as follows:

- The command **route set access-list <ACL_NAME>** pushes the prefixes from the ACL as static routes in the remote VPN peer routing table.
- The command **route set interface** is configured on the FlexVPN client and pushes its dynamically assigned IP address as static routes in the FlexVPN server routing table.
- The command **route accept any [tag|distance] <value>** accepts all routes from the remote VPN peer; optionally, you can assign a tag value or change the default administrative distance of 1.



VPN	Interop	Dynamic Routing	IPsec Routing	Spoke-spoke direct (shortcut)	Remote Access	Simple Failover	Source Failover	Config push	Per-peer config	Per-Peer QoS	Full AAA Management
Easy VPN	No	No	Yes	No	Yes	Yes	No	Yes	Yes	Yes	Yes
DMVPN	No	Yes	No	Yes	No	partial	No	No	No	group	No
Crypto Map	Yes	No	Yes	No	Yes	poor	No	No	No	No	No
Flex VPN	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Lab 1 IKEV2 Site-To-Site VPN (LAN-To-LAN) using SVTI & PSK



Basic Configuration

Internet

```
int e0/0
ip add 10.1.1.100 255.255.255.0
no sh
int e0/1
ip add 10.2.2.100 255.255.255.0
no sh
int e0/2
ip add 10.3.3.100 255.255.255.0
no sh
```

R1

```
ip domain-name cbtme.local
int e0/0
ip add 10.1.1.1 255.255.255.0
int loop 0
ip add 1.1.1.1 255.255.255.0
no sh
ip route 0.0.0.0 0.0.0.0 10.1.1.100
```

R2

```
ip domain-name cbtme.local
int e0/0
ip add 10.2.2.2 255.255.255.0
no sh
ip route 0.0.0.0 0.0.0.0 10.2.2.100

interface Loopback0
 ip address 2.2.2.2 255.255.255.0
```

R3 (will used later in further labs)

```
ip domain-name cbtme.local
int e0/0
ip add 10.3.3.3 255.255.255.0
no sh
ip route 0.0.0.0 0.0.0.0 10.3.3.100
```

Let's implement site to site IKEv2 between R1 and R2

Let's begin with R1 , First we need to configure IKEv2 proposal , policy , keyring , profile

You can use default crypto ikev2 proposal , crypto ikev2 policy which is part of what we call smart defaults

```
R1(config)#do sh crypto ikev2 proposal
 IKEv2 proposal: default
    Encryption : AES-CBC-256 AES-CBC-192 AES-CBC-128
    Integrity  : SHA512 SHA384 SHA256 SHA96 MD596
    PRF        : SHA512 SHA384 SHA256 SHA1 MD5
    DH Group   : DH_GROUP_1536_MODP/Group 5 DH_GROUP_1024_MODP/Group 2
```

```
R1(config)#do sh crypto ikev2 policy

 IKEv2 policy : default
     Match fvrf : any
     Match address local : any
     Proposal    : default
```

You still can modify above default settings or even create new ones using commands such as:

```
crypto ikev2 proposal Lab1
crypto ikev2 policy  Lab1
```

In this lab let's use the default for IKEv2 proposal & policy

Now we need to create keyring

```
crypto ikev2 keyring MYRING
 peer R2
  address 10.2.2.2
  identity fqdn R2.cbtme.local  < we can identify the peer with address , email , fqdn or key-id
  pre-shared-key local cisco
  pre-shared-key remote cisco
```

In case R1 will use same keyring above with another router such as R3 , we can type more than one peer in this keyring , like below :

```
crypto ikev2 keyring MYRING
 peer R2
  address 10.2.2.2
  identity fqdn R2.cbtme.local
  pre-shared-key local cisco
  pre-shared-key remote cisco

peer R3
  address 10.3.3.3
  identity fqdn R3.cbtme.local
  pre-shared-key local cisco
  pre-shared-key remote cisco
```

Now we need to create ikev2 profile

```
crypto ikev2 profile MYPROFILE
 match identity remote fqdn R2.cbtme.local   <Multiple “match identity” allowed
 match identity remote fqdn R3.cbtme.local
 identity local fqdn R1.cbtme.local    <Only one local identity allowed
 authentication remote pre-share  <Multiple remote methods allowed
 authentication local pre-share  <Only one local method allowed
 keyring local MYRING  < we type keyring aaa if we use AAA server
```

Second after we completed ikev2 components configuration we need to configure IPsec transform-set and IPsec profile

Again you can use default ones coming with ikev2 (smart defaults)

```
R1(config)#do sh crypto ipsec transform-set
Transform set default: { esp-aes esp-sha-hmac  }
   will negotiate = { Transport,  },
R1(config)#do sh crypto ipsec profile
IPSEC profile default
        Security association lifetime: 4608000 kilobytes/3600 seconds
        Responder-Only (Y/N): N
        PFS (Y/N): N
        Mixed-mode : Disabled
        Transform sets={
                default:  { esp-aes esp-sha-hmac  } ,
        }
```

let's use the default for both but we need only one extra command which is adding IKEv2 profile to IPsec profile

```
crypto ipsec profile default
set ikev2-profile MYPROFILE
exit
```

Notice i did not add transform-set since i am using the default one , and by default ipsec profile will use it

Finally time to configure our SVTI Tunnel

```
interface Tunnel1
 ip address 12.12.12.1 255.255.255.0
 ip mtu 1400
 ip tcp adjust-mss 1360
 tunnel source Ethernet0/0
 tunnel mode ipsec ipv4
 tunnel destination 10.2.2.2
 tunnel protection ipsec profile default

ip route 2.2.2.0 255.255.255.0 tunnel 1
```

R2

```
crypto ikev2 keyring MYRING
 peer R1
  address 10.1.1.1
  identity fqdn R1.cbtme.local
  pre-shared-key local cisco
  pre-shared-key remote cisco

crypto ikev2 profile MYPROFILE
 match identity remote fqdn R1.cbtme.local
 identity local fqdn R2.cbtme.local
 authentication remote pre-share
 authentication local pre-share
 keyring local MYRING
```

```
crypto ipsec profile default
set ikev2-profile MYPROFILE

interface Tunnel1
 ip address 12.12.12.2 255.255.255.0
 ip mtu 1400
 ip tcp adjust-mss 1360
 tunnel source Ethernet0/0
 tunnel mode ipsec ipv4
 tunnel destination 10.1.1.1
 tunnel protection ipsec profile default

ip route 1.1.1.0 255.255.255.0 tunnel1
```

```
R2#ping 1.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 5/5/6 ms

R2#sh crypto ikev2 sa
 IPv4 Crypto IKEv2  SA

Tunnel-id Local             Remote            fvrf/ivrf         Status
1         10.2.2.2/500      10.1.1.1/500        none/none          READY
      Encr: AES-CBC, keysize: 256, PRF: SHA512, Hash: SHA512, DH Grp:5, Auth sign: PSK, Auth
verify: PSK
      Life/Active Time: 86400/284 sec

 IPv6 Crypto IKEv2  SA

R2#sh crypto session
Crypto session current status

Interface: Tunnel1
Profile: MYPROFILE
Session status: UP-ACTIVE
Peer: 10.1.1.1 port 500
  Session ID: 1
  IKEv2 SA: local 10.2.2.2/500 remote 10.1.1.1/500 Active
  IPSEC FLOW: permit ip 0.0.0.0/0.0.0.0 0.0.0.0/0.0.0.0
        Active SAs: 2, origin: crypto map
```

Note:
If we want to change R1 to be like a hub and use DVTI , all what we need is remove the tunnel interface from R1 and add virtual-template interface

R1

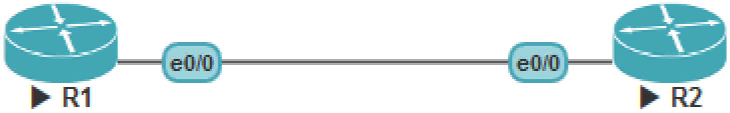
```
No int tunnel 1

int virtual-template 1 type tunnel
 ip add 12.12.12.1 255.255.255.0
tunnel source e0/0
tunnel mode ipsec ipv4
tunnel protection ipsec profile default
```

And in ikev2 profile we bind the virtual-template interface to it by adding the following line

```
crypto ikev2 profile MYPROFILE
virtual-template 1
```

Lab 2 IKEV2 Site-To-Site VPN (LAN-To-LAN) using SVTI & PSK



This Lab is similar to lab 1 but I would like to add more explanations to IKEv2 Components

Basic Configuration

R1

```
int e0/0
ip add 10.1.1.1 255.255.255.0
no sh
int loop 0
ip add 1.1.1.1 255.255.255.0
```

R2

```
int e0/0
ip add 10.1.1.2 255.255.255.0
no sh
int loop 0
ip add 2.2.2.2 255.255.255.0
```

R1

Define IKEv2 Proposal

```
crypto ikev2 proposal IKEPROP
encr aes-cbc-192
group 14
integrity sha512
```

An IKEv2 proposal consists of transforms which are used in the negotiation of IKE SAs, in IKE_SA_INIT exchange. proposal must have at least encr algorithm , integrity algorithm , dh algorithm configured

Define IKEv2 Policies

```
crypto ikev2 policy IKEPOL
proposal IKEPROP
```

An IKEv2 Policy contains IKEv2 Proposals (defined in above step) which are used to negotiate the Encryption Algorithm, Integrity Algorithm, PRF Algorithms, and Diffie-Hellman (DH) Group in IKE_SA_INIT exchange.

Define IKEv2 Keyring

```
crypto ikev2 keyring R1_TO_R2_K
peer R2
address 10.1.1.2
pre-shared-key local cisco123
pre-shared-key remote cisco123
```

An IKEv2 keyring consists of preshared keys associated with an IKEv2 profile. Authentication is performed by Pre-Shared Keys defined inside an IKEv2 keyring.)

Define IKEv2 Profiles

```
crypto ikev2 profile R1_TO_R2_P
match identity remote address 10.1.1.2
authentication  local pre-share
authentication remote pre-share
keyring local R1_TO_R2_K
```

IKEv2 Profiles are similar to IKEv1 ISAKMP Profile

A Transform Set is used to define how the data traffic between IPSec peers is going to be protected in Child Tunnel (IPSec Tunnel)

Define Transform Sets

```
crypto ipsec transform-set R1_R2 esp-aes 192 esp-sha512-hmac
```

Define IPSec Profile

```
crypto ipsec profile default
set ikev2-profile R1_TO_R2_P
set transform-set R1_R2
```

Configure Tunnel Interface Static VTI

```
int tunnel 10
ip address 192.168.100.1 255.255.255.0
tunnel source e0/0
tunnel dest 10.1.1.2
tunnel mode ipsec ipv4
tunnel protection ipsec profile default

router eigrp 100
no auto
network 1.1.1.1 0.0.0.0
network 192.168.100.1 0.0.0.0
```

R2

```
crypto ikev2 proposal IKEPROP
encr aes-cbc-192
group 14
integrity sha512

crypto ikev2 policy IKEPOL
proposal IKEPROP

crypto ikev2 keyring R1_TO_R2_K
peer R1
address 10.1.1.1
pre-shared-key local cisco123
pre-shared-key remote cisco123

crypto ikev2 profile R1_TO_R2_P
match identity remote address 10.1.1.1
authentication  local pre-share
authentication remote pre-share
keyring local R1_TO_R2_K

crypto ipsec transform-set R1_R2 esp-aes 192 esp-sha512-hmac

crypto ipsec profile default
set ikev2-profile R1_TO_R2_P
set transform-set R1_R2
```

```
int tunnel 10
ip address 192.168.100.2 255.255.255.0
tunnel source e0/0
tunnel dest 10.1.1.1
tunnel mode ipsec ipv4
tunnel protection ipsec profile default

router eigrp 100
no auto
network 2.2.2.2 0.0.0.0
network 192.168.100.2 0.0.0.0
```

Verification

R1#ping 2.2.2.2 source loopback 0

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
Packet sent with a source address of 1.1.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 6/6/6 ms
```

R1#sh crypto ikev2 sa

```
 IPv4 Crypto IKEv2  SA

Tunnel-id Local            Remote           fvrf/ivrf        Status
1         10.1.1.1/500     10.1.1.2/500     none/none        READY
      Encr: AES-CBC, keysize: 192, PRF: SHA512, Hash: SHA512, DH Grp:14, Auth sign: PSK, Auth
verify: PSK
      Life/Active Time: 86400/16 sec
```

R1#sh crypto ikev2 session

```
 IPv4 Crypto IKEv2 Session

Session-id:1, Status:UP-ACTIVE, IKE count:1, CHILD count:1

Tunnel-id Local            Remote           fvrf/ivrf        Status
1         10.1.1.1/500     10.1.1.2/500     none/none        READY
      Encr: AES-CBC, keysize: 192, PRF: SHA512, Hash: SHA512, DH Grp:14, Auth sign: PSK, Auth
verify: PSK
      Life/Active Time: 86400/20 sec
Child sa: local selector  0.0.0.0/0 - 255.255.255.255/65535
          remote selector 0.0.0.0/0 - 255.255.255.255/65535
          ESP spi in/out: 0x4989DA62/0x47550147
```

R1#sh ip route eigrp

```
      2.0.0.0/24 is subnetted, 1 subnets
D        2.2.2.0 [90/27008000] via 192.168.100.2, 00:00:36, Tunnel10
```

R1#sh crypto ikev2 proposal default

```
 IKEv2 proposal: default
     Encryption : AES-CBC-256 AES-CBC-192 AES-CBC-128
     Integrity  : SHA512 SHA384 SHA256 SHA96 MD596
     PRF        : SHA512 SHA384 SHA256 SHA1 MD5
     DH Group   : DH_GROUP_1536_MODP/Group 5 DH_GROUP_1024_MODP/Group 2
R1#sh crypto ikev2 policy default
 IKEv2 policy : default
      Match fvrf : any
      Match address local : any
      Proposal    : default
```

R1#sh crypto ipsec transform-set deafult

```
Transform set deafult not found
R1#sh crypto ipsec transform-set R1_R2
{ esp-192-aes esp-sha512-hmac  }
   will negotiate = { Tunnel,  },
```

R1#sh crypto ikev2 profile

```
IKEv2 profile: R1_TO_R2_P
 Ref Count: 5
 Match criteria:
  Fvrf: global
  Local address/interface: none
  Identities:
   address 10.1.1.2 255.255.255.255
  Certificate maps: none
 Local identity: none
 Remote identity: none
 Local authentication method: pre-share
 Remote authentication method(s): pre-share
 EAP options: none
 Keyring: R1_TO_R2_K
 Trustpoint(s): none
 Lifetime: 86400 seconds
 DPD: disabled
 NAT-keepalive: disabled
 Ivrf: none
 Virtual-template: none
 mode auto: none
 AAA AnyConnect EAP authentication mlist: none
 AAA EAP authentication mlist: none
 AAA Accounting: none
 AAA group authorization: none
 AAA user authorization: none
```

R1#sh crypto ikev2 session detailed

```
 IPv4 Crypto IKEv2 Session

Session-id:1, Status:UP-ACTIVE, IKE count:1, CHILD count:1

Tunnel-id Local           Remote          fvrf/ivrf        Status
1         10.1.1.1/500    10.1.1.2/500      none/none        READY
      Encr: AES-CBC, keysize: 192, PRF: SHA512, Hash: SHA512, DH Grp:14, Auth sign: PSK, Auth
verify: PSK
      Life/Active Time: 86400/212 sec
      CE id: 1001, Session-id: 1
      Status Description: Negotiation done
      Local spi: A1796B77930C73C6       Remote spi: 3CF871A8BB8DF61A
      Local id: 10.1.1.1
      Remote id: 10.1.1.2
      Local req msg id:  0            Remote req msg id:  2
      Local next msg id: 0            Remote next msg id: 2
      Local req queued:  0            Remote req queued:  2
      Local window:      5            Remote window:      5
      DPD configured for 0 seconds, retry 0
      Fragmentation not  configured.
      Extended Authentication not configured.
      NAT-T is not detected
      Cisco Trust Security SGT is disabled
      Initiator of SA : No
Child sa: local selector  0.0.0.0/0 - 255.255.255.255/65535
          remote selector 0.0.0.0/0 - 255.255.255.255/65535
          ESP spi in/out: 0x4989DA62/0x47550147
          AH spi in/out: 0x0/0x0
```

Lab 3 IKEV2 Site-To-Site VPN (LAN-To-LAN) using Crypto Map & PSK

Same Lab 2 topology
(This lab just to practice more, cisco is not recommend using crypto map any more)

Basic Configuration
R1

```
config t
hostname R1
int e0/0
ip add 10.1.1.1 255.255.255.0
no sh
int loop 0
ip add 1.1.1.1 255.255.255.0
ip route 2.2.2.0 255.255.255.0 10.1.1.2
```

R2

```
config t
hostname R2
int e0/0
ip add 10.1.1.2 255.255.255.0
no sh
int loop 0
ip add 2.2.2.2 255.255.255.0
ip route 1.1.1.0 255.255.255.0 10.1.1.1
```

We will Disable any default IKEv2/IPsec settings which relate to my configuration.
We will Use a crypto-map based configuration on R1 and R2
Encrypted traffic will be between both routers loopback 0

R1

```
no crypto ikev2 policy default
no crypto ikev2 proposal default
no crypto ipsec transform-set default
!
crypto ikev2 proposal IKEPROP
encr aes-cbc-192
group 14
integrity sha512
!
crypto ikev2 policy IKEPOL
proposal IKEPROP

crypto ikev2 keyring R1_TO_R2_K
peer R2
address 10.1.1.2
pre-shared-key cisco
!
```

```
crypto ikev2 profile R1_TO_R2_P
match identity remote address 10.1.1.2 255.255.255.255
authentication  local pre-share
authentication remote pre-share
keyring local R1_TO_R2_K
!
```

Define Crypto ACL to identify IPSec secured traffic

Crypto ACL is just an ACL created using normal ACL syntax, with permit or deny statements. Crypto ACLs are not used to permit or deny traffic similar to normal ACLs. In Crypto ACL, a permit statement is used to identify the traffic which is to be secured using IPSec and a deny statement is used to identify the traffic whish doesn't need to be secured. Here we are using "named extended access lists".)

```
ip access-list extended R1_R2_ACL
 permit ip host 1.1.1.1 host 2.2.2.2

crypto ipsec transform-set R1_R2 esp-aes 192 esp-sha512-hmac
```

Define Crypto Maps

Crypto Maps are used to connect all the pieces of IPSec configuration together. A Crypto Map consists of one or more entries. A Crypto Map is made up of Crypto ACL, Transform Set, Remote Peer, the lifetime of the data connections etc.

```
crypto map VPN 100 ipsec-isakmp
 set peer 10.1.1.2
 match address R1_R2_ACL
set transform-set R1_R2
 set ikev2-profile R1_TO_R2_P
 set security-association lifetime seconds 7200
set pfs group14
!
```

Activate Crypto Maps by applying the Crypto Map to Router's Interface

```
interface e0/0
 crypto map VPN
```

R2

```
no crypto ikev2 policy default
no crypto ikev2 proposal default
no crypto ipsec transform-set default
!
!
crypto ikev2 proposal IKEPROP
encr aes-cbc-192
group 14
integrity sha512
!
crypto ikev2 policy IKEPOL
proposal IKEPROP

crypto ikev2 keyring R1_TO_R2_K
peer R1
address 10.1.1.1
pre-shared-key cisco
!
crypto ikev2 profile R1_TO_R2_P
match identity remote address 10.1.1.1 255.255.255.255
authentication  local pre-share
authentication remote pre-share
keyring local R1_TO_R2_K
!
ip access-list extended R1_R2_ACL
 permit ip host 2.2.2.2 host 1.1.1.1
!
crypto ipsec transform-set R1_R2 esp-aes 192 esp-sha512-hmac
!
crypto map VPN 100 ipsec-isakmp
 set peer 10.1.1.1
 match address R1_R2_ACL
set transform-set R1_R2
 set ikev2-profile R1_TO_R2_P
 set security-association lifetime seconds 7200
set pfs group14
!
interface e0/0
 crypto map VPN
```

Troubleshooting
During configuration I faced issue not made vpn to come up
One of the best command to use in this case is :

```
debug crypto ikev2 packet
```

Then try to use the traffic which should be encrypted to trigger the VPN such as :

```
ping 2.2.2.2 source loopback 0
```

You will notice the following message:

```
*Sep  3 17:18:10.452: IKEv2-PAK:(SESSION ID = 1,SA ID = 1):Next payload: ENCR, version: 2.0
Exchange type: IKE_AUTH, flags: RESPONDER MSG-RESPONSE Message id: 1, length: 96
Payload contents:
 NOTIFY(AUTHENTICATION_FAILED)  Next payload: NONE, reserved: 0x0, length: 8
    Security protocol id: IKE, spi size: 0, type: AUTHENTICATION_FAILED
```

```
u all
```

My authentication has something wrong , after checking up I found space in 'cisco ' word used as pre shared key in R1 , so i will rewrite it again:

```
crypto ikev2 keyring R1_TO_R2_K
peer R2
address 10.1.1.2
pre-shared-key cisco
```

Verification

```
R1#ping 2.2.2.2 source loopback 0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
Packet sent with a source address of 1.1.1.1
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 8/8/10 ms
```

```
R1#sh crypto ikev2 sa
 IPv4 Crypto IKEv2  SA

Tunnel-id Local                 Remote                fvrf/ivrf            Status
1         10.1.1.1/500          10.1.1.2/500          none/none            READY
      Encr: AES-CBC, keysize: 192, PRF: SHA512, Hash: SHA512, DH Grp:14, Auth sign: PSK, Auth
verify: PSK
      Life/Active Time: 86400/49 sec

 IPv6 Crypto IKEv2  SA
```

Lab 4 IKEV2 Site-To-Site VPN (LAN-To-LAN) using Crypto Map & DVTI & PSK

Same Lab 2 topology

Here I will again Disable any default IKEv2/IPsec settings which relate to my configuration.
I will Use crypto map based configuration for R1 and a DVTI based configuration on R2
R1 will be initiator and R2 will be responder.
Encrypted traffic will be between both routers loopback 0

R1

```
config t
hostname R1
int e0/0
ip add 10.1.1.1 255.255.255.0
no sh
int loop 0
ip add 1.1.1.1 255.255.255.0
ip route 2.2.2.0 255.255.255.0 10.1.1.2
```

R2

```
config t
hostname R2
int e0/0
ip add 10.1.1.2 255.255.255.0
no sh
int loop 0
ip add 2.2.2.2 255.255.255.0
ip route 1.1.1.0 255.255.255.0 10.1.1.1
```

R1

```
no crypto ikev2 policy default
no crypto ikev2 proposal default
no crypto ipsec transform-set default

crypto ikev2 proposal IKEPROP
encr aes-cbc-192
group 14
integrity sha512
!
crypto ikev2 policy IKEPOL
proposal IKEPROP

crypto ikev2 keyring R1_TO_R2_K
peer R2
address 10.1.1.2
pre-shared-key cisco
!
```

```
crypto ikev2 profile R1_TO_R2_P
match identity remote address 10.1.1.2 255.255.255.255
authentication  local pre-share
authentication remote pre-share
keyring local R1_TO_R2_K
!
ip access-list extended R1_R2_ACL
 permit ip host 1.1.1.1 host 2.2.2.2
!
crypto ipsec transform-set R1_R2 esp-aes 192 esp-sha512-hmac
!
crypto map VPN 100 ipsec-isakmp
 set peer 10.1.1.2
 match address R1_R2_ACL
set transform-set R1_R2
 set ikev2-profile R1_TO_R2_P
 set security-association lifetime seconds 7200
set pfs group14
!
interface e0/0
 crypto map VPN
```

R2

```
no crypto ikev2 policy default
no crypto ikev2 proposal default
no crypto ipsec transform-set default
no crypto ipsec profile default

crypto ikev2 proposal IKEPROP
encr aes-cbc-192
group 14
integrity sha512
!
crypto ikev2 policy IKEPOL
proposal IKEPROP
!
crypto ikev2 keyring R1_TO_R2_K
peer R1
address 10.1.1.1
pre-shared-key cisco
!
crypto ikev2 profile R1_TO_R2_P
match identity remote address 10.1.1.1 255.255.255.255
authentication  local pre-share
authentication remote pre-share
keyring local R1_TO_R2_K
 virtual-template 1
!
```

```
crypto ipsec transform-set R1_R2 esp-aes 192 esp-sha512-hmac
!
crypto ipsec profile R1_R2_IPSEC
 set ikev2-profile R1_TO_R2_P
 set transform-set R1_R2
!
interface Virtual-Template1 type tunnel
 ip unnumbered e0/0
 tunnel source e0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile R1_R2_IPSEC
```

Verification

```
R1#ping 2.2.2.2 source loopback 0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
Packet sent with a source address of 1.1.1.1
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 8/8/10 ms
```

Notice once ping start from R1 ,the virtual access 1 interface automatically come up in R2
R2

```
*Sep  3 18:27:32.886: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state to up
R2#
```

```
R1#sh crypto ikev2 sa
 IPv4 Crypto IKEv2  SA

Tunnel-id Local           Remote          fvrf/ivrf        Status
1         10.1.1.1/500    10.1.1.2/500    none/none        READY
      Encr: AES-CBC, keysize: 192, PRF: SHA512, Hash: SHA512, DH Grp:14, Auth sign: PSK, Auth verify: PSK
      Life/Active Time: 86400/55 sec

 IPv6 Crypto IKEv2  SA
```

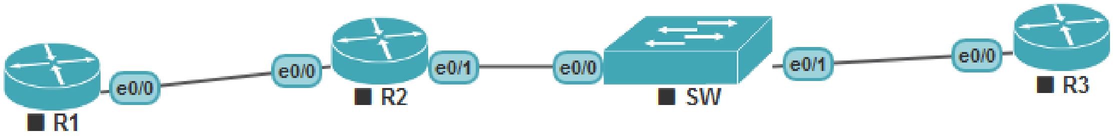
```
R2#sh crypto ikev2 sa
 IPv4 Crypto IKEv2  SA

Tunnel-id Local           Remote          fvrf/ivrf        Status
1         10.1.1.2/500    10.1.1.1/500    none/none        READY
      Encr: AES-CBC, keysize: 192, PRF: SHA512, Hash: SHA512, DH Grp:14, Auth sign: PSK, Auth verify: PSK
      Life/Active Time: 86400/70 sec

 IPv6 Crypto IKEv2  SA
```

Lab 5 Flex VPN Server/client

Flex VPN Server/client used as replacement for Easy VPN Server/Remote



- R1 will be Flex VPN client and R2 will be Flex VPN server:
- We will use the default IKEv2/IPsec configurations.
- Authentication will be PSK authentication with the string MYFLEXKEY.
- Flex VPN server should assign an IPv4 address from subnet 136.1.12.0/24 to the Flex VPN client.
- R1 should install a default route and that R2 installs a route for internal network through the IPsec tunnel, both with a tag value of 12.

- Make sure to match the following output by using an IKEv2 ID of FQDN:

```
R1#show crypto ikev2 session detailed | i cbtme.local
    Local id: flexclient. cbtme.local
    Remote id: flexserver. cbtme.local
```

Basic Configuration

R1

```
ip domain-name cbtme.local
int E0/0
ip add 10.1.1.1 255.255.255.0
int loop 0
ip add 1.1.1.1 255.255.255.0
```

R2

```
ip domain-name cbtme.local
int E0/0
ip add 10.1.1.2 255.255.255.0
int loop 0
ip add 2.2.2.2 255.255.255.0
int E0/1
ip add 192.168.100.2 255.255.255.0
```

R3 internal network

```
ip domain-name cbtme.local
int e0/0
ip add 192.168.100.1 255.255.255.0
ip route 0.0.0.0 0.0.0.0 192.168.100.2
```

Flex VPN Configuration

R1

```
default crypto ikev2 policy
default crypto ikev2 proposal
default crypto ipsec transform-set
default crypto ipsec profile
default crypto ikev2 authorization policy
!
aaa new-model
aaa authorization network IKE_LIST local
!
interface Tunnel12
 ip address negotiated
 tunnel source E0/0
 tunnel destination dynamic
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile default
!
crypto ikev2 client flexvpn R1_R2_CLIENT
 peer 1 10.1.1.2
 connect auto
 client connect Tunnel12
!
ip access-list standard PROTECTED_ACL
 permit 192.168.100.0 0.0.0.255
!
crypto ikev2 authorization policy default
 route accept any tag 12
 route set interface
 route set access-list PROTECTED_ACL
!
crypto ikev2 keyring R1_R2_KEYRING
 peer R2
  address 10.1.1.2
  pre-shared-key local MYFLEXKEY
  pre-shared-key remote MYFLEXKEY
!
crypto ikev2 profile R1_R2_PROFILE
 match identity remote fqdn flexserver.cbtme.local
 identity local fqdn flexclient.cbtme.local
 authentication local pre-share
 authentication remote pre-share
 keyring local R1_R2_KEYRING
 aaa authorization group psk list IKE_LIST default
!
!
crypto ipsec profile default
 set ikev2-profile R1_R2_PROFILE
```

R2

```
default crypto ikev2 policy
default crypto ikev2 proposal
default crypto ipsec transform-set
default crypto ipsec profile
default crypto ikev2 authorization policy
!
aaa new-model
aaa authorization network IKE_LIST local
!
ip local pool IKE_POOL 192.168.100.100 192.168.100.254
!
ip access-list standard PROTECTED_ACL
 permit any
!
crypto ikev2 authorization policy default
 pool IKE_POOL
 route accept any tag 12
 no route set interface
 route set access-list PROTECTED_ACL
!
crypto ikev2 keyring R1_R2_KEYRING
 peer ALL
  address 0.0.0.0 0.0.0.0
  pre-shared-key local MYFLEXKEY
  pre-shared-key remote MYFLEXKEY
!
crypto ikev2 profile R1_R2_PROFILE
 match identity remote fqdn flexclient.cbtme.local
 identity local fqdn flexserver.cbtme.local
 authentication local pre-share
 authentication remote pre-share
 keyring local R1_R2_KEYRING
 aaa authorization group psk list IKE_LIST default
 virtual-template 1

crypto ipsec profile default
 set ikev2-profile R1_R2_PROFILE
!
interface Virtual-Template1 type tunnel
 ip unnumbered E0/0
 tunnel source E0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile default
```

Verification

Verify the default authorization policy; settings have been modified:

```
R1#show crypto ikev2 authorization policy default
```

Verify the flexvpn client connection status:

```
R1#show crypto ikev2 client flexvpn
```

Verify the flexvpn client tunnel interface:

```
R1#show interfaces tunnel12
```

Verify the IKEv2 session status:

```
R1#show crypto ikev2 session detailed
R2#show crypto ikev2 session detailed
```

Verify the installed IKEv2 routes:

```
R1#show ip route static | b Gateway
R1#show ip route 0.0.0.0
R2#show ip route static | b Gateway
R2#show ip route 136.1.11.0
```

Verify that traffic is encrypted through the IPsec tunnel:

```
R2#ping 192.168.100.1 source E0/0
R2#ping 136.1.12.1 source gigabitEthernet0/0.22
```

```
R1#sh ip int br
Interface              IP-Address      OK? Method Status                Protocol
Ethernet0/0            10.1.1.1        YES manual up                    up
Loopback0              1.1.1.1         YES manual up                    up
Tunnel12               192.168.100.100 YES manual up                    up
```

```
R1#sh ip route
S*    0.0.0.0/0 is directly connected, Tunnel12
      1.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        1.1.1.0/24 is directly connected, Loopback0
L        1.1.1.1/32 is directly connected, Loopback0
      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.1.1.0/24 is directly connected, Ethernet0/0
L        10.1.1.1/32 is directly connected, Ethernet0/0
      192.168.100.0/32 is subnetted, 1 subnets
C        192.168.100.100 is directly connected, Tunnel12
```

```
R2#sh ip route
S*    0.0.0.0/0 is directly connected, Virtual-Access1
      2.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        2.2.2.0/24 is directly connected, Loopback0
L        2.2.2.2/32 is directly connected, Loopback0
      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
```

```
C       10.1.1.0/24 is directly connected, Ethernet0/0
L       10.1.1.2/32 is directly connected, Ethernet0/0
     192.168.100.0/24 is variably subnetted, 3 subnets, 2 masks
C       192.168.100.0/24 is directly connected, Ethernet0/1
L       192.168.100.2/32 is directly connected, Ethernet0/1
S       192.168.100.100/32 is directly connected, Virtual-Access1
```

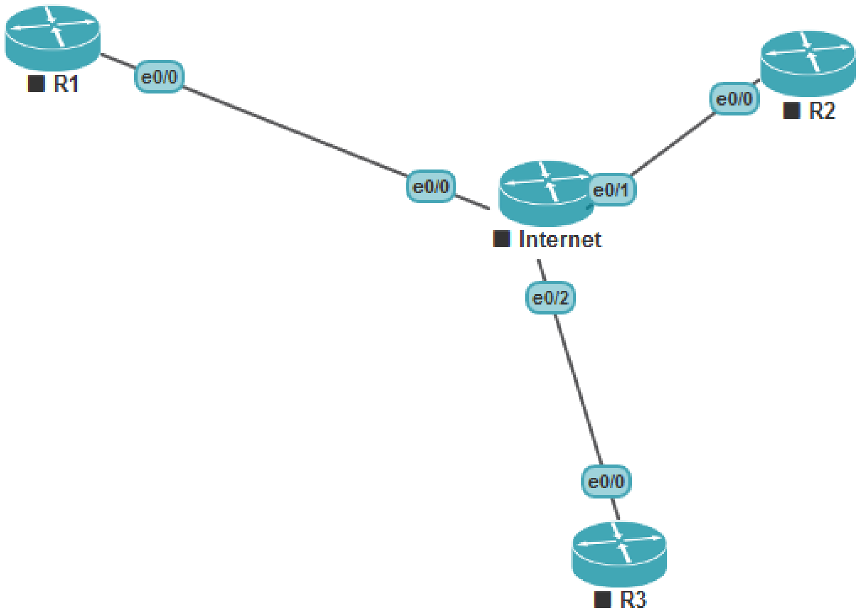
```
R2#ping 192.168.100.100 source e0/0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.100.100, timeout is 2 seconds:
Packet sent with a source address of 10.1.1.2
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 6/7/8 ms
```

```
R1#ping 192.168.100.2 source tunnel12
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.100.2, timeout is 2 seconds:
Packet sent with a source address of 192.168.100.100
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 7/7/8 ms
```

```
R2#show crypto ipsec sa | i #pkts
    #pkts encaps: 10, #pkts encrypt: 10, #pkts digest: 10
    #pkts decaps: 30, #pkts decrypt: 30, #pkts verify: 30
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
```

Lab 6 Flex VPN DMVPN (Flex VPN Spoke to Spoke) with PSK

DMVPN can be implemented with IKEv2 using the same configuration as in IKEv1, except that IKEv1 is replaced with IKev2 but when use Flex VPN to secure DMVPN ,Cisco added DMVPN functionality and changed both the configuration (to map it to the FlexVPN command set) and the functionality (to further optimize it).



Let me summarize first the new commands we are going to use :

Hub

int virtual-template 1 type tunnel **< used to communicate with spokes**
ip nhrp network-id 1
ip nhrp **redirect < means if Spoke1 send info to Spoke2 , tell spoke 1 how to reach Spoke 2 directly**

when spoke 1 R2 want to talk to spoke 2 R3 , the Hub with help of NHRP will be able to tell spoke 1 how to reach spoke 2 directly

Spokes

int tunnel 1 **< used to communicate with hub**
ip nhrp network-id 1 **< should be same id number used in Hub**
ip nhrp redirect
ip nhrp shortcut virtual-template 1

int virtual-template 1 type tunnel **< used to communicate with spokes**
ip nhrp network-id 1
ip nhrp redirect
ip nhrp shortcut virtual-template 1

As you can see in spokes we will have two interfaces

- **Tunnel interface SVTI to communicate with hub**
- **Virtual-template interface DVTI to communicate with other spokes , DVTI will help us to communicate each other spoke using separate virtual-access interfaces**

Notice that ip address for tunnel and virtual-template interface will be the same and can even be getting from hub itself if you like using the following commands:

R1 Hub

```
aaa new-model
aaa authorization network default local

ip local pool FlexSpokes 172.16.0.100 172.16.0.200

crypto ikev2 authorization policy default
 pool FlexSpokes
route set interface
route set access-list HUB-ACL

crypto ikev2 profile MYPROFILE
aaa authorization group cert list default default
```

R2 Spoke

```
int tunnel 1
ip address nego

int virtual-template 1 type tunnel
ip unnumbered tunnel1
```

Note: In below lab i will not use this pool option , but you can practice it by yourself later when you complete lab 6 with me

Basic Configuration

Same basic ip address configuration we used in previous lab

R1

```
ip domain-name cbtme.local
int e0/0
ip add 10.1.1.1 255.255.255.0
ip route 0.0.0.0 0.0.0.0 10.1.1.100
int loop 100
ip add 100.1.1.1 255.255.255.0
```

R2

```
ip domain-name cbtme.local
int e0/0
ip add 10.2.2.2 255.255.255.0
ip route 0.0.0.0 0.0.0.0 10.2.2.100
int loop 100
ip add 100.2.2.2 255.255.255.0
```

R3

```
ip domain-name cbtme.local
int e0/0
ip add 10.3.3.3 255.255.255.0
no sh
ip route 0.0.0.0 0.0.0.0 10.3.3.100
int loop 100
ip add 100.3.3.3 255.255.255.0
```

R1 Hub

```
aaa new-model
aaa authorization network default local

crypto ikev2 authorization policy default
route set interface
route set access-list HUB-ACL  < list of network that is reachable

crypto ikev2 keyring MYRING
peer R3
address 10.3.3.3
identity fqdn R3.cbtme.local
pre-shared-key local cisco
pre-shared-key remote cisco
peer R2
address 10.2.2.2
identity fqdn R2.cbtme.local
pre-shared-key local cisco
pre-shared-key remote cisco
exit
```

```
crypto ikev2 profile MYPROFILE
identity local fqdn R1.cbtme.local
match identity remote fqdn domain cbtme.local
authentication local pre-share
authentication remote pre-share
keyring local MYRING
aaa authorization group psk list default default
virtual-template 1
exit

crypto ipsec profile default
set ikev2-profile MYPROFILE
exit

int loop 0
ip add 172.16.1.1 255.255.255.0

int virtual-template 1 type tunnel
 ip unnumbered Loopback0
tunnel source e0/0
ip nhrp network-id 1
ip nhrp redirect
tunnel protection ipsec profile default

router eigrp 100
no auto
network 100.1.1.1 0.0.0.0
network 172.16.1.1 0.0.0.0
```

R2 Spoke 1

```
aaa new-model
aaa authorization network default local

crypto ikev2 authorization policy default
route set interface
route set access-list spoke-ACL

ip access-list stand spoke-ACL
permit 100.2.0.0 0.255.255.255

crypto ikev2 keyring MYRING
peer R3
address 10.3.3.3
identity fqdn R3.cbtme.local
pre-shared-key local cisco
pre-shared-key remote cisco
peer R1
address 10.1.1.1
```

```
identity fqdn R1.cbtme.local
pre-shared-key local cisco
pre-shared-key remote cisco
exit

crypto ikev2 profile MYPROFILE
identity local fqdn R2.cbtme.local
match identity remote fqdn domain cbtme.local
authentication local pre-share
authentication remote pre-share
keyring local MYRING
aaa authorization group psk list default default
virtual-template 1
exit

crypto ipsec profile default
set ikev2-profile MYPROFILE
exit

int loop 0
ip add 172.16.1.2 255.255.255.0

int virtual-template 1 type tunnel
 ip  unnumbered loopback0
tunnel source e0/0
ip nhrp network-id 1
ip nhrp shortcut virtual-template 1
ip nhrp redirect
tunnel protection ipsec profile default

interface Tunnel1
 ip unnumbered Loopback0
tunnel source e0/0
tunnel dest 10.1.1.1
ip nhrp network-id 1
ip nhrp shortcut virtual-template 1
ip nhrp redirect
tunnel protection ipsec profile default

router eigrp 100
no auto
network 100.2.2.2 0.0.0.0
network 172.16.1.2 0.0.0.0
```

R3 Spoke 2

```
aaa new-model
aaa authorization network default local

crypto ikev2 authorization policy default
route set interface
route set access-list spoke2-ACL

ip access-list stand spoke2-ACL
permit 100.3.0.0 0.255.255.255

crypto ikev2 keyring MYRING
peer R2
address 10.2.2.2
identity fqdn R2.cbtme.local
pre-shared-key local cisco
pre-shared-key remote cisco
peer R1
address 10.1.1.1
identity fqdn R1.cbtme.local
pre-shared-key local cisco
pre-shared-key remote cisco
exit

crypto ikev2 profile MYPROFILE
identity local fqdn R3.cbtme.local
match identity remote fqdn domain cbtme.local
authentication local pre-share
authentication remote pre-share
keyring local MYRING
aaa authorization group psk list default default
virtual-template 1
exit

crypto ipsec profile default
set ikev2-profile MYPROFILE
exit

int loop 0
ip add 172.16.1.3 255.255.255.0

int virtual-template 1 type tunnel
 ip  unnumbered loopback0
tunnel source e0/0
ip nhrp network-id 1
ip nhrp shortcut virtual-template 1
ip nhrp redirect
tunnel protection ipsec profile default
```

```
interface Tunnel1
 ip unnumbered Loopback0
tunnel source e0/0
tunnel dest 10.1.1.1
ip nhrp network-id 1
ip nhrp shortcut virtual-template 1
ip nhrp redirect
tunnel protection ipsec profile default

router eigrp 100
no auto
network 100.3.3.3 0.0.0.0
network 172.16.1.3 0.0.0.0
```

Verification

R2#sh crypto session

```
Crypto session current status

Interface: Tunnel1
Profile: MYPROFILE
Session status: UP-ACTIVE
Peer: 10.1.1.1 port 500
  Session ID: 1
  IKEv2 SA: local 10.2.2.2/500 remote 10.1.1.1/500 Active
  IPSEC FLOW: permit 47 host 10.2.2.2 host 10.1.1.1
        Active SAs: 2, origin: crypto map
```

R1#show crypto session

```
Crypto session current status

Interface: Virtual-Access1
Profile: MYPROFILE
Session status: UP-ACTIVE
Peer: 10.2.2.2 port 500
  Session ID: 1
  IKEv2 SA: local 10.1.1.1/500 remote 10.2.2.2/500 Active
  IPSEC FLOW: permit 47 host 10.1.1.1 host 10.2.2.2
        Active SAs: 2, origin: crypto map
```

```
R1#sh ip route eigrp
Gateway of last resort is 10.1.1.100 to network 0.0.0.0

      100.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
D        100.2.2.0/24 [90/27008000] via 172.16.1.2, 00:00:24, Virtual-Access1
```

```
R2#sh ip route eigrp
Gateway of last resort is 10.2.2.100 to network 0.0.0.0

      100.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
D        100.1.1.0/24 [90/27008000] via 172.16.1.1, 00:01:03, Tunnel1
```

```
R3#ping 10.2.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

```
R3#sh crypto session
Crypto session current status

Interface: Tunnel1
Profile: MYPROFILE
Session status: UP-ACTIVE
Peer: 10.1.1.1 port 500
  Session ID: 1
  IKEv2 SA: local 10.3.3.3/500 remote 10.1.1.1/500 Active
  IPSEC FLOW: permit 47 host 10.3.3.3 host 10.1.1.1
        Active SAs: 2, origin: crypto map
```

```
R3#sh crypto ikev2 sa
 IPv4 Crypto IKEv2  SA

Tunnel-id Local                 Remote                fvrf/ivrf            Status
1           10.3.3.3/500          10.1.1.1/500          none/none            READY
      Encr: AES-CBC, keysize: 256, PRF: SHA512, Hash: SHA512, DH Grp:5, Auth sign: PSK, Auth
verify: PSK
      Life/Active Time: 86400/67 sec

 IPv6 Crypto IKEv2  SA
```

```
R1#ping 10.2.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

R1#sh crypto session

```
Crypto session current status

Interface: Virtual-Access1
Profile: MYPROFILE
Session status: UP-ACTIVE
Peer: 10.2.2.2 port 500
  Session ID: 1
  IKEv2 SA: local 10.1.1.1/500 remote 10.2.2.2/500 Active
  IPSEC FLOW: permit 47 host 10.1.1.1 host 10.2.2.2
        Active SAs: 2, origin: crypto map

Interface: Virtual-Access2
Profile: MYPROFILE
Session status: UP-ACTIVE
Peer: 10.3.3.3 port 500
  Session ID: 2
  IKEv2 SA: local 10.1.1.1/500 remote 10.3.3.3/500 Active
  IPSEC FLOW: permit 47 host 10.1.1.1 host 10.3.3.3
        Active SAs: 2, origin: crypto map
```

R1#sh ip eigrp nei

```
EIGRP-IPv4 Neighbors for AS(100)
H   Address              Interface           Hold Uptime   SRTT   RTO  Q  Seq
                                 (sec)        (ms)       Cnt Num
1   172.16.1.3            Vi2                14 00:02:52   10  1470  0  10
0   172.16.1.2            Vi1                10 00:02:59   18  1470  0  12
```

R3#ping 100.2.2.2 source 10.3.3.3

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 100.2.2.2, timeout is 2 seconds:
Packet sent with a source address of 10.3.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 7/7/8 ms
```

R3#ping 100.1.1.1 source 10.3.3.3

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 100.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 10.3.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 7/9/14 ms
```

```
R3#sh crypto ipsec sa

interface: Tunnel1
    Crypto map tag: Tunnel1-head-0, local addr 10.3.3.3

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (10.3.3.3/255.255.255.255/47/0)
   remote ident (addr/mask/prot/port): (10.1.1.1/255.255.255.255/47/0)
   current_peer 10.1.1.1 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 88, #pkts encrypt: 88, #pkts digest: 88
    #pkts decaps: 65, #pkts decrypt: 65, #pkts verify: 65
```

```
R2#sh crypto ikev2 authorization policy default
 IKEv2 Authorization Policy : default
  route set interface
  route set acl: spoke-ACL
  route accept any tag : 1 distance : 1
```

```
R2#sh access-lists
Standard IP access list spoke-ACL
    10 permit 100.0.0.0, wildcard bits 0.255.255.255
```

```
R2#ping 100.3.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 100.3.3.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 6/7/8 ms
```

```
R2#traceroute 100.3.3.3
Type escape sequence to abort.
Tracing the route to 100.3.3.3
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.1.3 11 msec 11 msec *
```

```
R2#sh ip int br
Interface                  IP-Address      OK? Method Status                Protocol
Ethernet0/0                  10.2.2.2        YES NVRAM  up                    up
Loopback0                    172.16.1.2      YES NVRAM  up                    up
Loopback100                  100.2.2.2       YES NVRAM  up                    up
Tunnel1                    172.16.1.2      YES TFTP   up                    up
Virtual-Access1              172.16.1.2      YES unset  up                    up
Virtual-Template1            172.16.1.2      YES unset  up                    down
```

```
R2# sh ip nhrp shortcut
100.3.3.0/24 via 172.16.1.3
  Virtual-Access1 created 00:03:11, expire 01:56:48
  Type: dynamic, Flags: router rib nho
  NBMA address: 10.3.3.3
172.16.1.3/32 via 172.16.1.3
  Virtual-Access1 created 00:03:11, expire 01:56:48
  Type: dynamic, Flags: router nhop rib nho
  NBMA address: 10.3.3.3
```

```
R2#sh dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        T1 - Route Installed, T2 - Nexthop-override
        C - CTS Capable
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
==========================================================================

Interface: Virtual-Access1, IPv4 NHRP Details
Type:Unknown, NHRP Peers:1,

 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----
     2 10.3.3.3             172.16.1.3    UP 00:04:07   DT2
                            172.16.1.3    UP 00:04:07   DT2
```

As we can see Spoke 1 R2 can Talk to Spoke 2 R3 directly , thanks to Hub and NHRP for making that happens
Notice I titled the lab Flex VPN DMVPN but actually we should call it Flex VPN Spoke to Spoke since this is not really pure DMVPN , It is how to use Flex VPN to do the same things DMVPN can do .

Virtual Template Lock

Effective with CSCtt26236, the virtual template lock allows you to modify or delete a virtual template of type tunnel only when the virtual template is not associated with any cloned virtual access interfaces. The virtual template lock prevents dynamic command updates from virtual templates to the cloned virtual access interfaces, which can cause instability in some scenarios.

If you try to modify or delete an active virtual template of type tunnel, the following error message appears:

```
Device(config)# interface virtual-template 1
 % Virtual-template config is locked, active vaccess present
```

Although the virtual template cannot be modified when the virtual template is associated with a virtual access interface, perform the following steps to modify an existing virtual template configuration:

- Configure a new virtual template interface. For more information, see “Configuring Dynamic IPsec Virtual Tunnel Interfaces.”

- Associate the new virtual template to the IKEv2 profile. For more information, see the Configuring IKEv2 Profile (Basic) module.

- Clear the active sessions using the clear crypto session command or wait for session termination. The new session will use the new virtual template.

EIGRP & IP Unnumbered

EIGRP behavior is changed by the ip unnumbered command. It disables checks for the same subnet while it establishes an EIGRP adjacency.

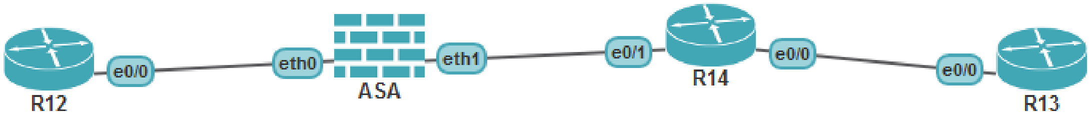
It is also important to remember that when you use DVTIs statically configured IP address on the virtual-template, it is not cloned to the virtual-access. This is why the ip unnumbered command is needed.

For FlexVPN, there is no need to use the ip unnumbered command when you use the negotiated address on the client. But, it is important to use it on the Hub when you use EIGRP. When you use the configuration mode for routing, EIGRP is not needed.

For SVTI, IPv6 uses link-local addresses for adjacency, and there is no need to use the ipv6 unnumbered command.

For DVTI, IPv6 cannot be configured manually. The ipv6 unnumbered command is recommended for the Hub, and the ipv6 address negotiated command is recommended on the Spoke.

Lab 7 Site to Site IKEv2 VPN ASA-IOS with PSK



Basic Configuration

R12

```
ip domain-name cbtme.com

int e0/0
ip add 10.12.12.12 255.255.255.0
int loop 0
ip add 12.12.12.12 255.255.255.0
ip route 0.0.0.0 0.0.0.0 10.12.12.100
```

ASA

```
domain-name cbtme.com
int e0
nameif inside
ip address 10.12.12.100 255.255.255.0
no sh
int e1
nameif outside
ip address 40.40.40.100 255.255.255.0
no sh
route inside 12.12.12.0 255.255.255.0 10.12.12.12
route outside 13.13.13.0 255.255.255.0 40.40.40.114
```

R14

```
ip domain-name cbtme.com
int e0/1
ip add 40.40.40.114 255.255.255.0
no sh
int e0/0
ip add 10.13.13.114 255.255.255.0
no sh
ip route 0.0.0.0 0.0.0.0 40.40.40.100
ip route 13.13.13.0 255.255.255.0 10.13.13.13
```

R13

```
ip domain-name cbtme.com
int e0/0
ip add 10.13.13.13 255.255.255.0
int loop 0
ip add 13.13.13.13 255.255.255.0
ip route 0.0.0.0 0.0.0.0 10.13.13.114
```

ASA does not have IKEv2 smart defaults, but as soon as an IKEv2 policy (which is the IKEv2 proposal from IOS) is created, the following default values are used, unless modified:

- **Encryption: 3des**
- **Integrity: sha1**
- **DH group: 2**
- **PRF: sha1**
- **Lifetime: 86400 seconds**

IOS IKEv2 Proposal = ASA IKEv2 Policy

For a newly created IKEv2 ipsec proposal on the ASA (which is the IPsec transform-set from IOS), if no encryption and integrity algorithms are specified, the following default values are used (the same as for the IKEv2 policy):

- Encryption: 3des
- Integrity: sha1

IOS IKEv2 Transform Set = ASA IKEv2 Proposal

Note: on IOS if the IKEv2 profile not attached to crypto map or IPsec profile , IKEv2 cannot be initiated so the router can act only as a responder

Now let's Configure IKEv2 IPsec between ASA4 and R2 as the following :

- Negotiate AES-256 and 3DES for encryption, SHA256 and MD5 for integrity, DH groups 5 and 2.
- Protect traffic between R2 and R1/R3 Loopback0 subnets, negotiate AES-256 and 3DES for encryption, SHA1 and MD5 for integrity.
- R2 should authenticate with a PSK of **r2psk** and ASA4 with a PSK of **asa4psk**.
- Limit the number of SAs to two and ensure that R2 generates a log message when the tunnel is UP/DOWN.

R14

```
crypto ikev2 proposal R14_ASA_PROPOSAL
 encryption aes-cbc-256 3des
 integrity sha256 md5
 group 5 2
!
crypto ikev2 limit max-sa 2
crypto logging ikev2
!
crypto ikev2 keyring R14_ASA_KEYRING
 peer ASA
  address 40.40.40.100
  pre-shared-key local r14psk
  pre-shared-key remote asapsk
```

```
!
crypto ikev2 profile R14_ASA_PROFILE
 match identity remote address 40.40.40.100
 authentication local pre-share
 authentication remote pre-share
 keyring local R14_ASA_KEYRING
!
ip access-list extended R14_ASA_ACL
 permit ip host 13.13.13.13 host 12.12.12.12

!
crypto ipsec transform-set auda esp-aes 256 esp-sha-hmac
crypto ipsec transform-set auda2 esp-3des esp-md5-hmac
!
crypto map VPN 100 ipsec-isakmp
 set peer 40.40.40.100
 match address R14_ASA_ACL
 set transform-set auda auda2
 set ikev2-profile  R14_ASA_PROFILE
!
interface e0/1
 crypto map VPN
```

ASA

```
crypto ikev2 enable outside
crypto ikev2 policy 100
 encryption aes-256 3des
 integrity sha256 md5
 group 5 2
 prf sha256 md5
!
crypto ikev2 limit max-sa 2
!
access-list R14_ASA_ACL permit ip host 12.12.12.12 host 13.13.13.13
!
crypto ipsec ikev2 ipsec-proposal R14_ASA_PROPOSAL
 protocol esp encryption aes-256 3des
 protocol esp integrity sha-1 md5
!
tunnel-group 40.40.40.114 type ipsec-l2l
tunnel-group 40.40.40.114 ipsec-attributes
 ikev2 local-authentication pre-shared-key asapsk
 ikev2 remote-authentication pre-shared-key r14psk
!
crypto map VPN 100 set peer 40.40.40.114
crypto map VPN 100 match address R14_ASA_ACL
crypto map VPN 100 set ikev2 ipsec-proposal R14_ASA_PROPOSAL
crypto map VPN interface outside
```

Verification

R12#ping 13.13.13.13 source loopback0

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 13.13.13.13, timeout is 2 seconds:
Packet sent with a source address of 12.12.12.12
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/5/17 ms
```

ASA(config)# sh crypto ikev2 sa

```
IKEv2 SAs:

Session-id:1, Status:UP-ACTIVE, IKE count:1, CHILD count:1

Tunnel-id                 Local                Remote     Status       Role
 10464467     40.40.40.100/500      40.40.40.114/500      READY    INITIATOR
      Encr: AES-CBC, keysize: 256, Hash: SHA256, DH Grp:5, Auth sign: PSK, Auth verify: PSK
      Life/Active Time: 86400/76 sec
Child sa: local selector  12.12.12.12/0 - 12.12.12.12/65535
          remote selector 13.13.13.13/0 - 13.13.13.13/65535
          ESP spi in/out: 0x390820da/0xe3ce587e
ASA(config)# sh cry
ASA(config)# sh crypto ipsec sa | i #pk
      #pkts encaps: 9, #pkts encrypt: 9, #pkts digest: 9
      #pkts decaps: 5, #pkts decrypt: 5, #pkts verify: 5
      #pkts compressed: 0, #pkts decompressed: 0
      #pkts not compressed: 9, #pkts comp failed: 0, #pkts decomp failed: 0
ASA(config)#
```

Good Luck
CCIE & CCSI: Yasser Auda
https://www.facebook.com/YasserRamzyAuda
https://learningnetwork.cisco.com/people/yasserramzy/content
https://www.youtube.com/user/yasserramzyauda