# Configure Site-to-Site FlexVPN Tunnel With a Peer With Dynamic IP Address

## Contents



## Introduction

This document describes how to configure a FlexVPN site-to-site VPN tunnel between 2 Cisco Routers when the remote peer has a dynamic IP address.

## Prerequisites

### Requirements

Cisco recommends that you have knowledge of these topics:

- FlexVPN
- IKEv2 Protocol

### Components Used

The information in this document is based on these software and hardware versions:

- CSR1000V device
- Cisco IOS® XE Software, Version 17.3.4

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

# Background Information

## Network Diagram


*Topology for Dynamic Peer*

The topology in this example shows a Cisco router and another Cisco router that has a dynamic IP address on its public-facing interface.

# Configurations

This section describes how to configure the Site-to-Site FlexVPN tunnel on a Cisco router when the remote peer uses a dynamic IP address.

In this configuration example, the authentication method used is Pre-Shared-Key (PSK) however, Public key Infrastructure (PKI) can be used as well.

## Configuration on Headquarter Router

In this example, the IKEv2 Smart Defaults from the router have been used. The IKEv2 Smart Defaults feature minimizes the FlexVPN configuration by covering most of the use cases. IKEv2 smart defaults can be customized for specific use cases, though this is not recommended. The smart defaults includes the IKEv2 Authorization policy, IKEv2 proposal, IKEv2 policy, Internet Protocol Security (IPsec) Profile, and IPsec transform set.

To review the default values in your device, you can run the commands listed below.

- show crypto ikev2 authorization policy default
- show crypto ikev2 proposal default
- show crypto ikev2 policy default
- show crypto ipsec profile default
- show crypto ipsec transform-set default

Step 1 Configure IKEv2 keyring.

- In this case, since the headquarters router does not know the peer ip due to it being dynamic the identity it matches to any ip address.
- Remote and local keys are also configured.
- It is recommended to have strong keys to avoid any vulnerability.

```
crypto ikev2 keyring FLEXVPN_KEYRING
```

```
 peer spoke
  address 0.0.0.0 0.0.0.0
  pre-shared-key local Cisco123
  pre-shared-key remote Cisco123
```

Step 2 Configure the Authentication, Authorization and Accounting (AAA) model.

- This creates the management framework for the users that can connect for this instance.
- Since the connection negotiation is initiated from this device the model references its local database to determine the authorize users.

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

Step 3 Configure the IKEv2 profile.

- Given that the remote peer IP address is dynamic you cannot use a specific IP address to identity the peer.
- You can however, identify the remote peer by domain, FQDN  or Key-id defined on the peer device.

- The authentication, Authorization and Accounting (AAA) group needs to be added for the authorization method of the profile specifying PSK is the method used.
- If the authentication method is PKI here it is specified as cert instead of PKI .
- Since the objective is to create a Dynamic Virtual Tunnel Interface (dVTI) this profile is linked to a virtual template

```
crypto ikev2 profile FLEXVPN_PROFILE
 match identity remote key-id Peer123
 identity local address 172.16.1.1
 authentication remote pre-share
 authentication local pre-share
 keyring local FLEXVPN_KEYRING
 aaa authorization group psk list FLEXVPN default
 virtual-template 1
```

Step 4 Configure the IPsec profile.

- A custom IPsec profile can be configured if you do not use the default profile.
- The IKEv2 profile created in Step 3 is mapped to this IPsec profile.

```
crypto ipsec profile defualt
 set ikev2-profile FLEXVPN_PROFILE
```

Step 5 Configure the Loopback interface and Virtual Template Interface.

- Since the remote device has a dynamic IP address, a dVTI needs to be created from a template.

- This virtual template interface is a configuration template from which dynamic Virtual-Access interfaces are created.

```
interface Loopback1
 ip address 192.168.1.1 255.255.255.0
```

```
interface Virtual-Template1 type tunnel
 ip unnumbered Loopback1
 tunnel protection ipsec profile default
```

## Branch Router Configuration

For the branch router, configure the IKEv2 Keyring, AAA model, IPsec profile, and IKEv2 profile as indicated on the previous steps with the necessary configuration changes and the ones described next:

1. Configure the local identity which is sent to the headquarters router as identifier.

```
crypto ikev2 profile FLEXVPN_PROFILE
 identity local key-id Peer123
 match identity remote address 172.16.1.1
 authentication remote pre-share
 authentication local pre-share
 keyring local FLEXVPN_KEYRING
 aaa authorization group psk list FLEXVPN default
```

Step 5 Configure Static Virtual Tunnel Interface.

- Given that the IP adress for the Headquarter router is know and does not change, a Static VTI interface is configured.

```
interface Tunnel0
 ip address 192.168.1.10 255.255.255.0
 tunnel source GigabitEthernet0
 tunnel destination 172.16.1.1
 tunnel protection ipsec profile default
```

## Routing Configuration

In this example, routing is defined during the establishment of the IKEv2 Security Association (SA) with the configuration of an Access Control List. This defines the traffic to be sent over the VPN. You can also configure dynamic routing protocols, however it is not in the scope of this document.

Step 5. Define the ACL.

Headquarters Router:

```
ip access-list standard Flex-ACL
 permit 10.10.10.0 255.255.255.0
```

Branch Router:

```
ip access-list standard Flex-ACL
 permit 10.20.20.0 255.255.255.0
```

Step 6. Modify the IKEv2 Authorization Profiles on each router to set the ACL.

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

## Headquarter router complete configuration

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

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

crypto ikev2 keyring FLEXVPN_KEYRING
 peer spoke
  address 0.0.0.0 0.0.0.0
  pre-shared-key local Cisco123
  pre-shared-key remote Cisco123

crypto ikev2 profile FLEXVPN_PROFILE
 match identity remote key-id Peer123
 identity local address 172.16.1.1
 authentication remote pre-share
 authentication local pre-share
 keyring local FLEXVPN_KEYRING
 aaa authorization group psk list FLEXVPN default
 virtual-template 1

crypto ipsec profile default
 set ikev2-profile FLEXVPN_PROFILE

interface Loopback1
 ip address 192.168.1.1 255.255.255.0

interface Loopback10
 ip address 10.10.10.10 255.255.255.255
```

```
interface GigabitEthernet0
 ip address 172.16.1.1 255.255.255.0

interface Virtual-Template1 type tunnel
 ip unnumbered Loopback1
 tunnel protection ipsec profile default

ip access-list standard Flex-ACL
 5 permit 10.10.10.0 255.255.255.0
```

### Branch router complete configuration

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

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

crypto ikev2 keyring FLEXVPN_KEYRING
 peer HUB
  address 0.0.0.0 0.0.0.0
  pre-shared-key local Cisco123
  pre-shared-key remote Cisco123

crypto ikev2 profile FLEXVPN_PROFILE
 identity local key-id Peer123
 match identity remote address 172.16.1.1
 authentication remote pre-share
 authentication local pre-share
 keyring local FLEXVPN_KEYRING
 aaa authorization group psk list FLEXVPN default

crypto ipsec profile default
 set ikev2-profile FLEXVPN_PROFILE

interface Loopback20
 ip address 10.20.20.20 255.255.255.255

interface Tunnel0
 ip address 192.168.1.10 255.255.255.0
 tunnel source GigabitEthernet0
 tunnel destination 172.16.1.1
 tunnel protection ipsec profile default

interface GigabitEthernet0
 ip address dhcp
 negotiation auto

ip access-list standard Flex-ACL
 10 permit 10.20.20.0 255.255.255.0
```

# Verify

To verify the tunnel, you must verify Phase 1 and Phase 2 are up and working properly.

```
Headquarter#show crypto ikev2 sa detail
 IPv4 Crypto IKEv2  SA

Tunnel-id Local                 Remote                fvrf/ivrf            Status
1         172.16.1.1/500        172.16.2.1/500        none/none            READY
      Encr: AES-CBC, keysize: 256, PRF: SHA512, Hash: SHA512, DH Grp:19, Auth sign: PSK, Auth verify: P
      Life/Active Time: 86400/74645 sec
      CE id: 61256, Session-id: 1
      Status Description: Negotiation done
      Local spi: D5129F36B1180175       Remote spi: F9298874F90BFEC7
      Local id: 172.16.1.1
      Remote id: 172.16.2.1
      Local req msg id:  16             Remote req msg id:  31
      Local next msg id: 16             Remote next msg id: 31
      Local req queued:  16             Remote req queued:  31
      Local window:      5              Remote window:      5
      DPD configured for 0 seconds, retry 0
      Fragmentation not  configured.
      Dynamic Route Update: enabled
      Extended Authentication not configured.
      NAT-T is not detected
      Cisco Trust Security SGT is disabled
      Initiator of SA : No
      Remote subnets:                  -------------> This section shows the traffic to be routed acros
      192.168.1.10 255.255.255.255
      10.20.20.20 255.255.255.255

 IPv6 Crypto IKEv2  SA
```

Phase 2, Ipsec

```
Headquarter#show crypto ipsec sa

interface: Virtual-Access1
    Crypto map tag: Virtual-Access1-head-0, local addr 172.16.1.1

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (172.16.1.1/255.255.255.255/47/0)
   remote ident (addr/mask/prot/port): (172.16.2.1/255.255.255.255/47/0)
   current_peer 172.16.2.1 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 225, #pkts encrypt: 0, #pkts digest: 0
    #pkts decaps: 225, #pkts decrypt: 225, #pkts verify: 225
    #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.: 172.16.1.1, remote crypto endpt.: 172.16.2.1
     plaintext mtu 1458, path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0
     current outbound spi: 0xC124D7C1(3240417217)
     PFS (Y/N): N, DH group: none

     inbound esp sas:
```

```
     spi: 0xC2AADCAB(3265977515)
       transform: esp-aes esp-sha-hmac ,
       in use settings ={Transport, }
       conn id: 2912, flow_id: CSR:912, sibling_flags FFFFFFFF80000008, crypto map: Virtual-Access1-he
       sa timing: remaining key lifetime (k/sec): (4607993/628)
       IV size: 16 bytes
       replay detection support: Y
       Status: ACTIVE(ACTIVE)

    inbound ah sas:

    inbound pcp sas:

    outbound esp sas:
     spi: 0xC124D7C1(3240417217)
       transform: esp-aes esp-sha-hmac ,
       in use settings ={Transport, }
       conn id: 2911, flow_id: CSR:911, sibling_flags FFFFFFFF80000008, crypto map: Virtual-Access1-he
       sa timing: remaining key lifetime (k/sec): (4608000/628)
       IV size: 16 bytes
       replay detection support: Y
       Status: ACTIVE(ACTIVE)

    outbound ah sas:

    outbound pcp sas:
```

You also need to verify the Virtual Access interface is in UP state.

```
show interface Virtual-Access1
Virtual-Access2 is up, line protocol is up
  Hardware is Virtual Access interface
  Interface is unnumbered. Using address of Loopback1 (192.168.1.1)
  MTU 9934 bytes, BW 100 Kbit/sec, DLY 50000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL
  Tunnel vaccess, cloned from Virtual-Template1
  Vaccess status 0x4, loopback not set
  Keepalive not set
  Tunnel linestate evaluation up
  Tunnel source 172.16.1.1, destination 172.16.2.1
  Tunnel protocol/transport GRE/IP
    Key disabled, sequencing disabled
    Checksumming of packets disabled
  Tunnel TTL 255, Fast tunneling enabled
  Tunnel transport MTU 1434 bytes
  Tunnel transmit bandwidth 8000 (kbps)
  Tunnel receive bandwidth 8000 (kbps)
  Tunnel protection via IPSec (profile "default")
  Last input 20:53:34, output 20:53:34, output hang never
  Last clearing of "show interface" counters 20:55:43
  Input queue: 0/375/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     586 packets input, 149182 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicasts)
```

```
0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
15 packets output, 1860 bytes, 0 underruns
Output 0 broadcasts (0 IP multicasts)
0 output errors, 0 collisions, 0 interface resets
0 unknown protocol drops
0 output buffer failures, 0 output buffers swapped out
```

# Troubleshoot

This section describes how to troubleshoot the tunnel establishment

Complete these steps if the IKE negotiation fails:

1. Verify the current state with these commands:

- show crypto ikev2 sa
- show crypto ipsec sa
- show crypto session

2. Use these commands in order to debug the tunnel negotiation process:

- debug crypto ikev2
- debug crypto ipsec

# Related Information

- Cisco Technical Support & Downloads