

IKEV2 VRF Support

- Feature Summary and Revision History, on page 1
- Feature Description, on page 1
- How it works, on page 2
- Limitations, on page 2
- Configuring IKEv2 IPSec with VRF, on page 2

Feature Summary and Revision History

Summary Data

Applicable Product(s) or Functional Area	StarOS
Applicable Platform(s)	• ASR 5500 • VPC-DI
Feature Default	Not Applicable
Related Changes in This Release	Not Applicable
Related Documentation	Not Applicable

Revision History

Revision Details	Release
First introduced.	• 21.28.m6

Feature Description

In StarOS, without the IKEv2 VRF feature, only IPSec IKEv1 tunnels were VRF-aware, and where IKEv1 tunnels encrypted traffic originating from any VRF. Whereas the IPSec IKEv2 tunnels establish and encrypt

traffic only on default VRF. IKEv2 VRF feature supports IPSec IKEv2 tunnel establishment and traffic encryption on any VRF.

How it works

To support VRF for IKEv2 in the ASR5500 and VPC-DI environment, the decrypted packet needs to be processed in the right VRF so that it doesn't get discarded. The following function happens:

- The ACL gets hit for control and trigger packets for the first time.
- Sends packets to the IPSec and creates a tunnel.
- Starts the exchange of keys and a key pair establishes the tunnel.
- After the tunnel is established, that particular ipsecmgr flow DB entry gets deleted and creates a new entry. This ensures that the next packet passes to the crypto engine and gets forwarded. This is common to IPv4 and IPv6.

.

Limitations

Following are the limitations:

- The Key exchange and tunnel establishment occurs in the Default-VRF and not in the VRF whose traffic needs to be encrypted.
- The maximum number of IPSec ACLs per crypto map is seven. To support multiple IP chunks in an APN, multiple access-lists need to be configured. This might lead to multiple IPSec tunnels per enterprise VRF.
- Reconfiguring ACL rules that are corresponding to a crypto map requires reestablishment of the existing tunnels. This operation is disruptive for Uplink and Downlink subscriber traffics.

Configuring IKEv2 IPSec with VRF

Use the following sample configuration commands to configure IKEv2 IPSec with VRF. The following sample configuration shows how the loopback IP overlaps in the enterprise VRF and the Default-VRF to allow the exchange of keys from the Default-VRF but also allow crypto map to be applied in a VRF interface in the same time.

```
context ipsec-s
    ip vrf i-s
    #exit
    ip access-list boo
      permit ip host 2.1.1.1 host 2.2.1.1
    #exit
    crypto ipsec transform-set A-foo esp hmac sha1-96 cipher aes-cbc-128
      mode tunnel
    #exit
    ipsec transform-set B-foo
      hmac sha2-256-128
```

```
      group 14
    #exit
    ikev1 policy 1
    #exit
    ikev2-ikesa transform-set ikesa-foo
      group 14
      hmac sha2-256-128
      prf sha2-256
    #exit
    crypto map foo ikev2-ipv4
      match address boo
      authentication local pre-shared-key encrypted key
+B0bqvzhrkkwujr2kt37b0yxo4631silym4g2zn9r2rs0o7xrn3r4i09aexdk701t8d0cqt2ivg039da1267r6tcurpyk3qhdjbfwo7t6s

      authentication remote pre-shared-key encrypted key
+B0975tvzeoi0lg2zl78a17mnhv20yw3cesh97zi436qvsyoadulmh2pbgcnndjxchq0c3fn5p2i3y7b12uqc4bwsmi5x324ikw0wfzus8

      ikev2-ikesa transform-set list ikesa-foo
      ikev2-ikesa rekey
      payload foo-sa0 match ipv4
        ipsec transform-set list B-foo
        rekey keepalive
      #exit
      peer 5.2.1.1
    #exit
    interface ike
      ip address 192.168.110.120 255.255.255.0
    #exit
    interface iv1 loopback
      ip vrf forwarding i-s
      ip address 2.1.1.1 255.255.255.255
    #exit
    interface iv2 loopback
      ip vrf forwarding i-s
      ip address 5.1.1.1 255.255.255.255
      crypto-map foo
    #exit
    interface iv3 loopback
      ip address 5.1.1.1 255.255.255.255
    #exit
    subscriber default
    exit
    aaa group default
    #exit
    gtpp limit-secondary-rat-usage 32
    ip route 5.2.1.1 255.255.255.255 192.168.110.89 ike
  #exit
  port ethernet 1/10
    no shutdown
    vlan 110
      no shutdown
      bind interface ike ipsec-s
    #exit
  #exit
```