Configuring IKE-Enabled IPsec Tunnels

To securely transfer traffic from the Viptela overlay network to a service network, you can configure IPsec tunnels that run the Internet Key Exchange (IKE) protocol. IKE-enabled IPsec tunnels provide authentication and encryption to ensure secure packet transport.

You create an IKE-enabled IPsec tunnel by configuring an IPsec interface. IPsec interfaces are logical interfaces, and you configure them just like any other physical interface. You configure IKE protocol parameters on the IPsec interface, and you can configure other interface properties.

The Viptela software supports IKE, Version 1, as defined in RFC 2409, Internet Key Exchange, and IKE, Version 2, as defined in RFC 7296, Internet Key Exchange Protocol, Version 2.

One use for IPsec tunnels is to allow vEdge Cloud router VM instances running on Amazon AWS to connect to the Amazon virtual private cloud (VPC). You must configure IKE Version 1 on these routers.

Configure an IPsec Tunnel

To configure an IPsec tunnel interface for secure transport traffic from a service network, you create a logical IPsec interface:

```
vEdge(config)# vpn vpn-id interface ipsec number
vEdge(config-interface-ipsec)# ip address ipv4-prefix/length
vEdge(config-interface-ipsec)# (tunnel-source ip-address | tunnel-source-interface interface-name)
vEdge(config-interface-ipsec)# tunnel-destination ipv4-address
vEdge(config-interface-ipsec)# no shutdown
```

You can create the IPsec tunnel in any service VPN (VPN 1 through 65530, omitting 512).

The IPsec interface has a name in the format `ipsecnumber`, where `number` can be from 1 through 255.

Each IPsec interface must have an IPv4 address. This address must be a /30 prefix. All traffic in the VPN that is within this IPv4 prefix is directed to a physical interface in VPN 0 to be sent securely over an IPsec tunnel.

To configure the source of the IPsec tunnel on the local device, you can specify either the IP address of the physical interface (in the `tunnel-source` command) or the name of the physical interface (in the `tunnel-source-interface` command). Ensure that the physical interface is configured in VPN 0.

To configure the destination of the IPsec tunnel, specify the IP address of the remote device in the `tunnel-destination` command.

The combination of a source address (or source interface name) and a destination address defines a single IPsec tunnel. Only one IPsec tunnel can exist that uses a specific source address (or interface name) and destination address pair.

Enable IKE Version 1

When you create an IPsec tunnel on a vEdge router, IKE Version 1 is enabled by default on the tunnel interface. The following properties are also enabled by default for IKEv1:
• Authentication and encryption—AES-256 advanced encryption standard CBC encryption with the HMAC-SHA1 keyed-hash message authentication code algorithm for integrity
• Diffie-Hellman group number—16
• Rekeying time interval—4 hours
• SA establishment mode—Main

By default, IKEv1 uses IKE main mode to establish IKE SAs. In this mode, six negotiation packets are exchanged to establish the SA. To exchange only three negotiation packets, enabled aggressive mode:

```
vEdge(config)# vpn vpn-id interface ipsecnumber ike
vEdge(config-ike)# mode aggressive
```

By default, IKEv1 uses Diffie-Hellman group 16 in the IKE key exchange. This group uses the 4096-bit more modular exponential (MODP) group during IKE key exchange. You can change the group number to 2 (for 1024-bit MODP), 14 (2048-bit MODP), or 15 (3072-bit MODP):

```
vEdge(config)# vpn vpn-id interface ipsecnumber ike
vEdge(config-ike)# group number
```

By default, IKE key exchange uses AES-256 advanced encryption standard CBC encryption with the HMAC-SHA1 keyed-hash message authentication code algorithm for integrity. You can change the authentication to AES-128 CBC and the integrity checking to the HMAC-SHA1 keyed-has message authentication code algorithm:

```
vEdge(config)# vpn vpn-id interface ipsecnumber ike
vEdge(config-ike)# cipher-suite aes128-cbc-sha1
```

By default, IKE keys are refreshed every 1 hours (3600 seconds). You can change the rekeying interval to a value from 30 seconds through 14 days (1209600 seconds). It is recommended that the rekeying interval be at least 1 hour.

```
vEdge(config)# vpn vpn-id interface ipsecnumber ike
vEdge(config-ike)# rekey seconds
```

To force the generation of new keys for an IKE session, issue the `request ipsec ike-rekey` command.

```
vEdge(config)# vpn vpn-id interface ipsecnumber ike
```

For IKE, you can also configure preshared key (PSK) authentication:

```
vEdge(config)# vpn vpn-id interface ipsecnumber ike
vEdge(config-ike)# authentication-type pre-shared-key pre-shared-secret password
```

`password` is the password to use with the preshared key. It can be an ASCII or a hexadecimal string from 1 through 127 characters long.

If the remote IKE peer requires a local or remote ID, you can configure this identifier:

```
vEdge(config)# vpn vpn-id interface ipsecnumber ike authentication-type
vEdge(config-authentication-type)# local-id id
vEdge(config-authentication-type)# remote-id id
```

The identifier can be an IP address or any text string from 1 through 63 characters long. By default, the local ID is the tunnel's source IP address and the remote ID is the tunnel's destination IP address.
Enable IKE Version 2

When you configure an IPsec tunnel to use IKE Version 2, the following properties are also enabled by default for IKEv2:

- Authentication and encryption—AES-256 advanced encryption standard CBC encryption with the HMAC-SHA1 keyed-hash message authentication code algorithm for integrity
- Diffie-Hellman group number—16
- Rekeying time interval—4 hours

By default, IKEv2 uses Diffie-Hellman group 16 in the IKE key exchange. This group uses the 4096-bit more modular exponential (MODP) group during IKE key exchange. You can change the group number to 2 (for 1024-bit MODP), 14 (2048-bit MODP), or 15 (3072-bit MODP):

```
vEdge(config)# vpn vpn-id interface ipsecnumber ike
vEdge(config-ike)# group number
```

By default, IKE key exchange uses AES-256 advanced encryption standard CBC encryption with the HMAC-SHA1 keyed-hash message authentication code algorithm for integrity. You can change the authentication to AES-128 CBC and the integrity checking to the HMAC-SHA1 keyed-has message authentication code algorithm:

```
vEdge(config)# vpn vpn-id interface ipsecnumber ike
vEdge(config-ike)# cipher-suite aes128-cbc-sha1
```

By default, IKE keys are refreshed every 4 hours (14,400 seconds). You can change the rekeying interval to a value from 30 seconds through 14 days (1209600 seconds):

```
vEdge(config)# vpn vpn-id interface ipsecnumber ike
vEdge(config-ike)# rekey seconds
```

To force the generation of new keys for an IKE session, issue the `request ipsec ike-rekey` command.

For IKE, you can also configure preshared key (PSK) authentication:

```
vEdge(config)# vpn vpn-id interface ipsecnumber ike
vEdge(config-ike)# authentication-type pre-shared-key pre-shared-secret password
```

`password` is the password to use with the preshared key. It can be an ASCII or a hexadecimal string, or it can be an AES-encrypted key.

If the remote IKE peer requires a local or remote ID, you can configure this identifier:

```
vEdge(config)# vpn vpn-id interface ipsecnumber ike authentication-type
vEdge(config-ike)# local-id id
vEdge(config-ike)# remote-id id
```

The identifier can be an IP address or any text string from 1 through 64 characters long. By default, the local ID is the tunnel's source IP address and the remote ID is the tunnel's destination IP address.

Configure IPsec Tunnel Parameters

By default, the following parameters are used on the IPsec tunnel that carries IKE traffic:
• Authentication and encryption—AES-256 algorithm in GCM (Galois/counter mode)
• Rekeying interval—4 hours
• Replay window—32 packets

You can change the encryption on the IPsec tunnel to the AES-256 cipher in CBC (cipher block chaining mode, with HMAC-SHA1-96 keyed-hash message authentication or to null, to not encrypt the IPsec tunnel used for IKE key exchange traffic:

```
vEdge(config-interface-ipsecnumber)# ipsec
vEdge(config-ipsec)# cipher-suite (aes256-cbc-sha1 | aes256-gcm | null-sha1)
```

By default, IKE keys are refreshed every 4 hours (14,400 seconds). You can change the rekeying interval to a value from 30 seconds through 14 days (1209600 seconds):

```
vEdge(config-interface-ipsecnumber)# ipsec
vEdge(config-ipsec)# rekey seconds
```

To force the generation of new keys for an IPsec tunnel, issue the `request ipsec ipsec-rekey` command.

By default, perfect forward secrecy (PFS) is enabled on IPsec tunnels, to ensure that past sessions are not affected if future keys are compromised. PFS is available in Releases 17.2.3 and later. PFS forces a new Diffie-Hellman key exchange, by default using the 4096-bit Diffie-Hellman prime module group. You can change the PFS setting:

```
vEdge(config-interface-ipsecnumber)# ipsec
vEdge(config-ipsec)# perfect-forward-secrecy pfs-setting
```

`pfs-setting` can be one of the following:

- **group-2**—Use the 1024-bit Diffie-Hellman prime modulus group.
- **group-14**—Use the 2048-bit Diffie-Hellman prime modulus group.
- **group-15**—Use the 3072-bit Diffie-Hellman prime modulus group.
- **group-16**—Use the 4096-bit Diffie-Hellman prime modulus group. This is the default.
- **none**—Disable PFS.

By default, the IPsec replay window on the IPsec tunnel is 512 bytes. You can set the replay window size to 64, 128, 256, 512, 1024, 2048, or 4096 packets:

```
vEdge(config-interface-ipsecnumber)# ipsec
vEdge(config-ipsec)# replay-window number
```

---

**Modify IKE Dead-Peer Detection**

IKE uses a dead-peer detection mechanism to determine whether the connection to an IKE peer is functional and reachable. To implement this mechanism, IKE sends a Hello packet to its peer, and the peer sends an acknowledgment in response. By default, IKE sends Hello packets every 10 seconds, and after three unacknowledged packets, IKE declares the neighbor to be dead and tears down the tunnel to the peer. Thereafter, IKE periodically sends a Hello packet to determine if the peer is still alive.
packet to the peer, and re-establishes the tunnel when the peer comes back online.

You can change the liveness detection interval to a value from 0 through 65535 seconds, and you change change the number of retries to a value from 0 through 255:

```
  vEdge(config-interface-ipsecnumber)#  dead-peer-detection  seconds  retries  number
```

---

**Configure Other Interface Properties**

For IPsec tunnel interfaces, you can configure only the following additional interface properties:

```
  vEdge(config-interface-ipsec)#  mtu  bytes
  vEdge(config-interface-ipsec)#  tcp-mss-adjust  bytes
```

---

**Additional Information**

**Security Overview**