Create Umbrella SIG Manual Tunnel with Cisco Edge Devices

Contents

Introduction

This document describes how to build a CDFW Tunnel using a Cisco Edge Router running the 16.12 release in Umbrella SIG.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- The device must be fully configured and operational using the CLI-based templates before configuring the Umbrella SIG relevant parts mentioned later in this article. Only relevant items to the tunnel configuration are captured here.
- NAT must be configured in one or more of the transport VPN interfaces.
- The policy listed is a workaround until "allow-service ipsec" is added in a future release.

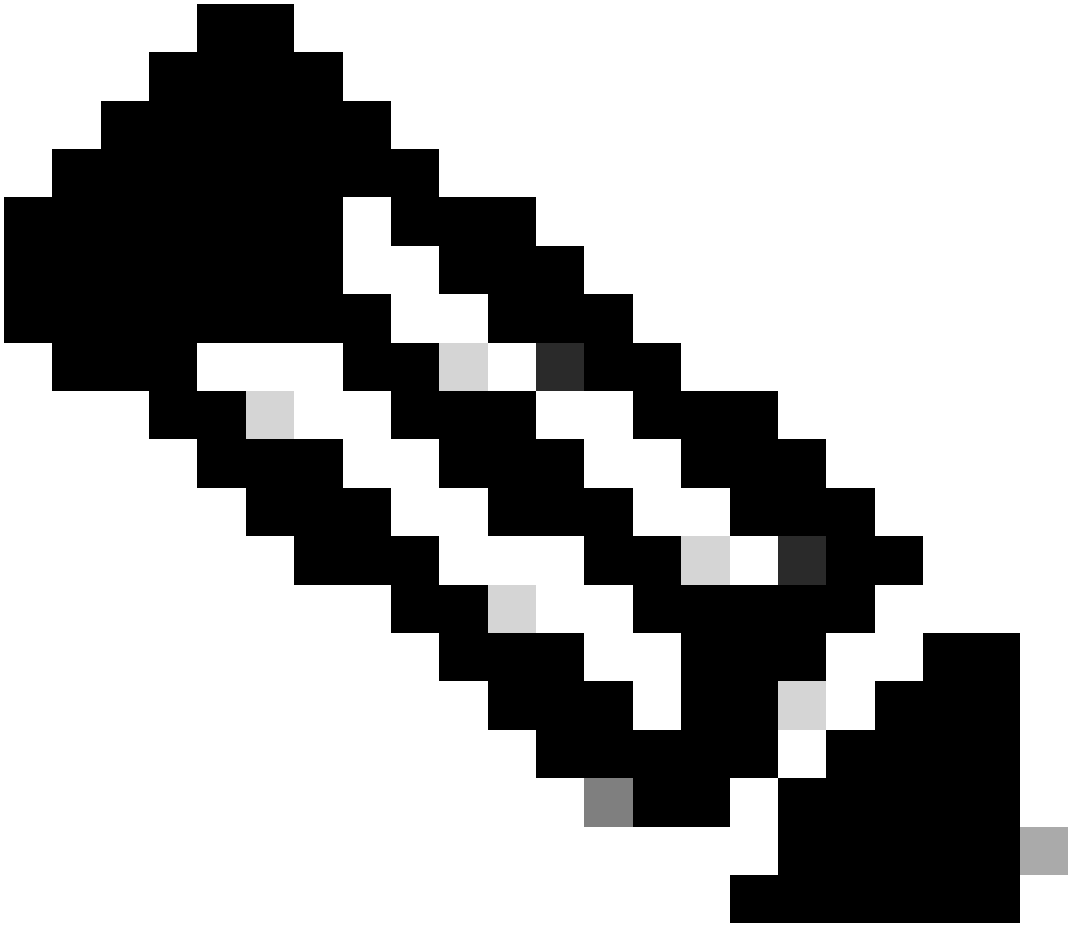
Components Used

The information in this document is based on Cisco Umbrella Secure Internet Gateway (SIG).

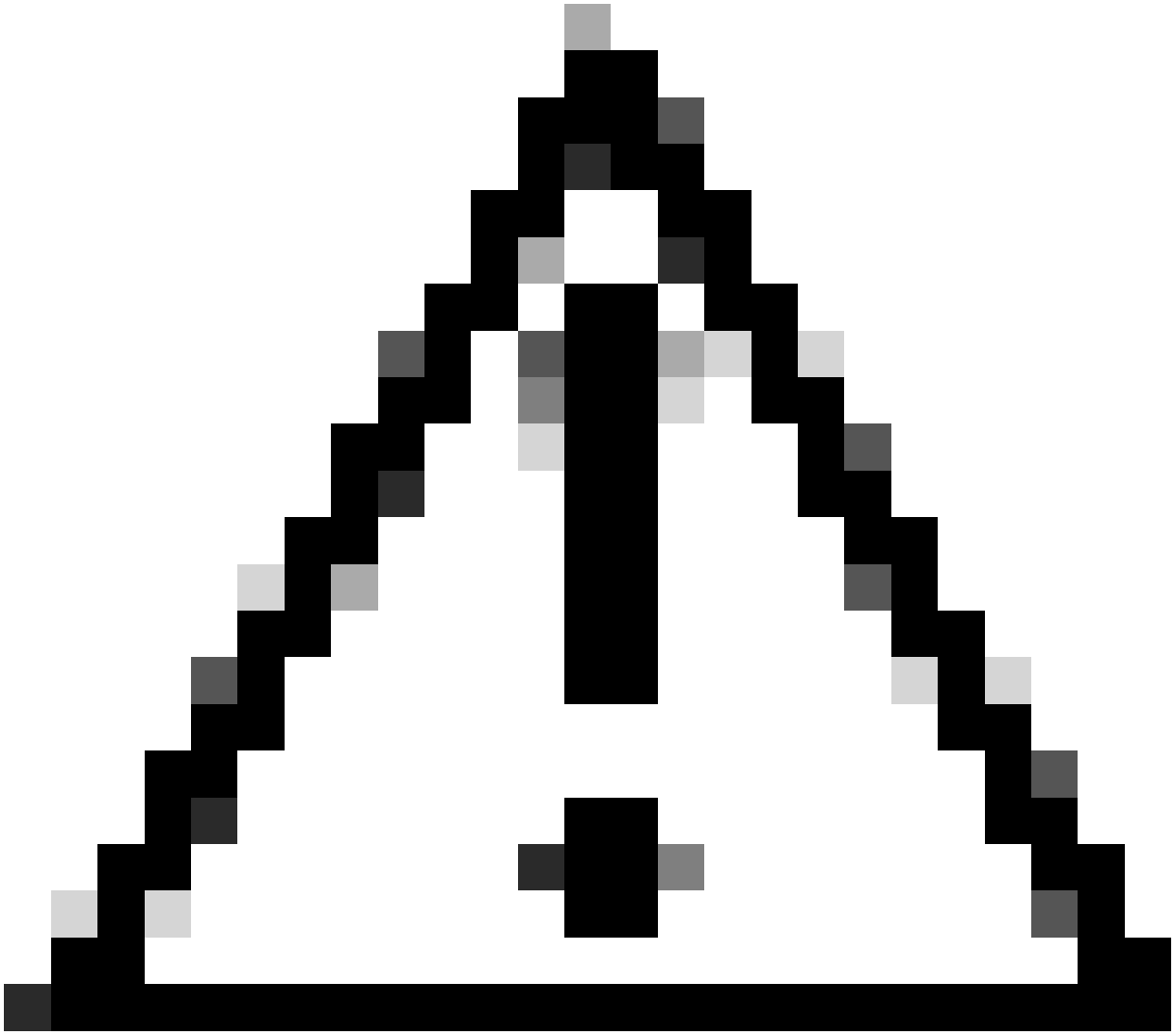
The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Overview

This article explains how to build a CDFW Tunnel using a Cisco Edge router (formerly Viptela cEdge) running the 16.12 release.



Note: The configuration template below is in INTENT based format, which is needed to create CLI-based tunnels in vManage. INTENT based format is similar to vEdge configuration format but there are some differences. A Feature template cannot effectively be used until 17.2.1 for cEdge, thus this example is using a CLI-based template.



Caution: This article was created to address the use case of sending corporate guest traffic through the Cisco Umbrella SIG solution. This how-to article uses CLI-based templates to override a limitation of Feature based templates in vManage.

Build the Manual Tunnel

1. Create a CDFW Tunnel in the Umbrella Dashboard.

2. Configure Viptela device template as you would normally configure for your environment.

3. Configure a SIG policy to allow ports UDP 500 and 4500 into transport interfaces. A

- CL_for_IKE_IPSec_tunnel is the ACL name that allows IPSEC traffic through the tunnel interface
- Optional: You can further restrict the ACL to only Umbrella SIG DCs. Read more in the Umbrella documentation.

```
access-list ACL_for_IKE_IPSec_tunnel
sequence 10
```

```
match
protocol 50
!
action accept
!
!
sequence 20
match
destination-port 4500 500
!
action accept
!
!
default-action drop
!
```

4. Apply the ACL to the tunnel interface that you are using.

```
sdwan
interface GigabitEthernet1
tunnel-interface
access-list ACL_for_IKE_IPSec_tunnel in
```

5. Configure the IPsec interface(s) in the transport VPN including required routes.

These variables are defined in the CLI config template after this list:

- `{transport_vpn_1}` is the network interface (typically the WAN interface) which establishes the IPSEC tunnel
- `{transport_vpn_ip_addr_prefix}` is the transport VPN that you assign. (for example, 1.1.1.0/24)
- `{ipsec__int_number}` is the IPSEC tunnel interface number (for example, the number 1 in interface "IPSEC1")
- `{ipsec_ip_addr_prefix}` is ip address and subnet defined for the IPSEC tunnel interface.
- `{transport_vpn_interface_1}` is the network interface (typically the WAN interface) that establishes the IPSEC tunnel. This is the same interface used in transport_vpn_1 variable.
- `{psk}` is the tunnel's pre-shared key value created in the Umbrella Dashboard's tunnels section.
- `{sig_fqdn}` is the tunnel's IKE ID created in the Umbrella Dashboard's tunnels section.
- `{sig_tunnel_dest_ip}` is the CDFW DC's IP the tunnel is connected to.

```
vpn 0
  interface {{transport_vpn_1}}
   ip address {{transport_vpn_ip_addr_prefix}}
   nat
    refresh bi-directional
   !
  mtu      1360
   no shutdown
  !
  interface ipsec{{ipsec__int_number}}
   ip address {{ipsec_ip_addr_prefix}}
   tunnel-source-interface {{transport_vpn_interface_1}}
   tunnel-destination      {{sig_tunnel_dest_ip}}
   ike
```

```
     version      2
     rekey        14400
     cipher-suite aes256-cbc-sha1
     group        14
     authentication-type
      pre-shared-key
       pre-shared-secret {{psk}}
       local-id          {{sig_fqdn}}
       remote-id         {{sig_tunnel_dest_ip}}
      !
     !
    !
    ipsec
     rekey                    3600
     replay-window            512
     cipher-suite             aes256-gcm
     perfect-forward-secrecy none
    !
    no shutdown
   !

ip ipsec-route 0.0.0.0/0 vpn 0 interface ipsec{{ipsec__int_number}}
```

For your reference, here is a sample configuration mentioned in steps 3-5:

```
access-list ACL_for_IKE_IPSec_tunnel
sequence 10
match
protocol 50
!
action accept
!
!
sequence 20
match
destination-port 4500 500
!
action accept
!
!
default-action drop
!
```

```
vpn 0
dns 208.67.222.222 primary
name VPN0
   interface GigabitEthernet4
    ip address 192.168.1.0/24
    nat
     refresh bi-directional
    !
   mtu      1360
    no shutdown
   !
   interface ipsec1
    ip address 10.10.10.1/30
```

```
    tunnel-source-interface GigabitEthernet4
    tunnel-destination      146.112.83.8
    ike
     version      2
     rekey        14400
     cipher-suite aes256-cbc-sha1
     group        14
     authentication-type
      pre-shared-key
       pre-shared-secret YourPreSharedKey
       local-id          YourTunnelID@umbrella.sig.cisco.com
       remote-id         146.112.83.8
      !
     !
    !
    ipsec
     rekey                    3600
     replay-window            512
     cipher-suite             aes256-gcm
     perfect-forward-secrecy none
    !
    no shutdown
   !
ip ipsec-route 0.0.0.0/0 vpn 0 interface ipsec1
```