Troubleshoot Common DMVPN Issues

Contents

Introduction

This document describes the most common solutions to Dynamic Multipoint VPN (DMVPN) problems.

Prerequisites

Requirements

Cisco recommends that you have knowledge of DMVPN configuration on Cisco IOS®routers.

Components Used

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

- Cisco IOS

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.

Conventions

Refer to Cisco Technical Tips Conventions for more information on document conventions.

Background Information

This document describes the most common solutions to Dynamic Multipoint VPN (DMVPN) problems. Many of these solutions can be implemented prior any in-depth troubleshoot of the DMVPN connection. This document is presented as a checklist of common procedures to try before you begin to troubleshoot a connection and call Cisco Technical Support.

For more information, refer to Dynamic Multipoint VPN Configuration Guide, Cisco IOS Release 15M&T .

Refer to Understand and Use Debug Commands to Troubleshoot IPsec to provide an explanation of common **debug** commands that are used to troubleshoot IPsec issues.

DMVPN Configuration Does Not Work

Problem

A recently configured or modified DMVPN solution does not work.

A current DMVPN configuration no longer works.

Solutions

This section contains solutions to the most common DMVPN problems.

These solutions (in no particular order) can be used as a checklist of items to verify or try before you troubleshoot in-depth :

- Common Issues
- Verify if Internet Security Association and Key Management Protocol (ISAKMP) Packets are Blocked at the Internet Service Provider (ISP).
- Verify if Generic Routing Encapsulation (GRE) works when the tunnel protection is removed.
- Next-Hop Resolution Protocol (NHRP) Registration Fails.
- Verify Whether the Lifetimes are Properly Configured.

- Verify Whether the Traffic flows in Only One Direction.
- Verify that the Routing Protocol Neighbor is Established.

Note: Before you begin, check the next steps:

1. Sync-up the timestamps between the hub and spoke
2. Enable **msec debug and log timestamps**:

   ```
   Router(config)#service timestamps debug datetime msec
   ```

   ```
   Router(config)#service timestamps log datetime msec
   ```

3. Enable **terminal exec prompt timestamp** for the debugging sessions:

   ```
   Router#terminal exec prompt timestamp
   ```

Note: This way, you can easily correlate the debug output with the show command output.

Common Issues

Verify the basic connectivity

1. Ping from the hub to the spoke with NBMA addresses and reverse.

   These pings must go directly out the physical interface, not through the DMVPN tunnel. Hopefully, there is not a firewall that blocks ping packets. If this does not work, check the routing and any firewalls between the hub and spoke routers.

2. Also, use **traceroute** to check the path that the encrypted tunnel packets take.
3. Use the **debug** and **show** commands to verify no connectivity:
   - **debug ip icmp**
   - **debug ip packet**

Note: The debug ip packet command generates a substantial amount of output and uses a substantial amount of system resources. This command must be used with caution in production networks. Always use with the access-list command. For more information on how to use the access-list with debug ip packet, refer to Troubleshoot with IP Access Lists.

Verify for Incompatible ISAKMP Policy

If the configured ISAKMP policies do not match the proposed policy by the remote peer, the router tries the default policy of 65535. If that does not match either, it fails the ISAKMP negotiation.

The **show crypto isakmp sa** command shows the ISAKMP SA to be in **MM_NO_STATE**, which mean the main-mode failed.

Verify for incorrect pre-shared key secret

If the pre-shared secrets are not the same on both sides, the negotiation fails.

The router returns the **sanity check failed** message.

Verify for Incompatible IPsec Transform Set

If the IPsec transform-set is not compatible or mismatched on the two IPsec devices, the IPsec negotiation fails.

The router returns the **atts not acceptable** message for the IPsec proposal.

Verify if ISAKMP Packets are Blocked at ISP

<#root>

```
Router#

show crypto isakmp sa



IPv4 Crypto ISAKMP SA
Dst                src                   state        conn-id      slot       status
172.17.0.1      172.16.1.1        MM_NO_STATE          0                0       ACTIVE
172.17.0.1      172.16.1.1        MM_NO_STATE          0                0       ACTIVE (deleted)
172.17.0.5        172.16.1.1        MM_NO_STATE      0              0       ACTIVE
172.17.0.5        172.16.1.1        MM_NO_STATE      0              0       ACTIVE (deleted)
```

The previous example shows the VPN tunnel flapping.

Further, check debug crypto isakmp to verify that the spoke router sends udp 500 packet:

<#root>

```
Router#

debug crypto isakmp
```

<#root>

```
04:14:44.450: ISAKMP:(0):Old State = IKE_READY
                       New State = IKE_I_MM1

04:14:44.450: ISAKMP:(0): beginning Main Mode exchange

04:14:44.450: ISAKMP:(0): sending packet to 172.17.0.1
              my_port 500 peer_port 500 (I) MM_NO_STATE
04:14:44.450: ISAKMP:(0):Sending an IKE IPv4 Packet.

04:14:54.450: ISAKMP:(0): retransmitting phase 1 MM_NO_STATE..
```

```
.
04:14:54.450: ISAKMP (0:0): incrementing error counter on sa,
              attempt 1 of 5: retransmit phase 1
04:14:54.450: ISAKMP:(0): retransmitting phase 1 MM_NO_STATE
04:14:54.450: ISAKMP:(0): sending packet to 172.17.0.1
              my_port 500 peer_port 500 (I) MM_NO_STATE
04:14:54.450: ISAKMP:(0):Sending an IKE IPv4 Packet.

04:15:04.450: ISAKMP:(0): retransmitting phase 1 MM_NO_STATE..

.
04:15:04.450: ISAKMP:(0): retransmitting phase 1 MM_NO_STATE...
04:15:04.450: ISAKMP (0:0): incrementing error counter on sa,
              attempt 2 of 5: retransmit phase 1
04:15:04.450: ISAKMP:(0): retransmitting phase 1 MM_NO_STATE
```

The previous debug output shows spoke router sends udp 500 packet in every 10 seconds.

Check with ISP to see if the spoke router is directly connected to the ISP router to make sure they allow udp 500 traffic.

After ISP allowed udp 500, add inbound ACL in egress interface, which is tunnel source to allow udp 500 to make sure udp 500 traffic comes into the router. Use the show access-list command to verify whether hit counts increment.

```
<#root>

Router#

show access-lists 101




Extended IP access list 101
    10 permit udp host 172.17.0.1 host 172.16.1.1 eq isakmp log (4 matches)
    20 permit udp host 172.17.0.5 host 172.16.1.1 eq isakmp log (4 matches)
    30 permit ip any any (295 matches)
```

Caution: Make sure you have ip any any allowed in your access-list. Otherwise, all other traffic can be blocked as an access-list applied inbound on the egress interface.

Verify if GRE Works When the Tunnel Protection is Removed

When DMVPN does not work, before you troubleshoot with IPsec, verify that the GRE tunnels work fine without IPsec encryption.

For more information, refer to How to Configure a GRE Tunnel.

NHRP Registration Fails

The VPN tunnel between hub and spoke is up, but unable to pass data traffic:

<#root>

```
Router#

show crypto isakmp sa

        dst             src             state          conn-id  slot   status
        172.17.0.1      172.16.1.1      QM_IDLE           1082    0    ACTIVE
```

<#root>

```
Router#

show crypto IPSEC sa

local  ident (addr/mask/prot/port): (172.16.1.1/255.255.255.255/47/0)
remote ident (addr/mask/prot/port): (172.17.0.1/255.255.255.255/47/0)

#pkts encaps: 154, #pkts encrypt: 154, #pkts digest: 154
#pkts decaps: 0, #pkts decrypt: 0, #pkts verify: 0

inbound esp sas:
spi: 0xF830FC95(4163959957)
outbound esp sas:
spi: 0xD65A7865(3596253285)

!--- !--- Output is truncated !---
```

It shows that return traffic does not come back from the other end of the tunnel.

Check NHS entry in the spoke router:

<#root>

```
Router#

show  ip nhrp nhs detail

Legend: E=Expecting replies, R=Responding
Tunnel0: 172.17.0.1  E  req-sent 0

 req-failed 30

 repl-recv 0
Pending Registration Requests:
Registration Request: Reqid 4371, Ret 64  NHS 172.17.0.1
```

It shows that the NHS request failed. To resolve this problem, make sure the configuration on the spoke router tunnel interface is correct.

Configuration example:

<#root>

```
interface Tunnel0
 ip address 10.0.0.9 255.255.255.0
 ip nhrp map 10.0.0.1 172.17.0.1
 ip nhrp map multicast 172.17.0.1


ip nhrp nhs 172.17.0.1


!--- !--- Output is truncated !---
```

Configuration example with the correct entry for the NHS server:

<#root>

```
interface Tunnel0
 ip address 10.0.0.9 255.255.255.0
 ip nhrp map 10.0.0.1 172.17.0.1
 ip nhrp map multicast 172.17.0.1


ip nhrp nhs 10.0.0.1


!--- !--- Output is truncated !---
```

Now, verify the NHS entry and IPsec encrypt/decrypt counters:

<#root>

```
Router#

show ip nhrp nhs detail

Legend: E=Expecting replies, R=Responding
Tunnel0:        10.0.0.1 RE  req-sent 4

req-failed 0

repl-recv 3 (00:01:04 ago)

Router#

show crypto IPSec sa


local  ident (addr/mask/prot/port): (172.16.1.1/255.255.255.255/47/0)
remote ident (addr/mask/prot/port): (172.17.0.1/255.255.255.255/47/0)

#pkts encaps: 121, #pkts encrypt: 121, #pkts digest: 121
#pkts decaps: 118, #pkts decrypt: 118, #pkts verify: 118

inbound esp sas:
spi: 0x1B7670FC(460747004)
```

```
outbound esp sas:
spi: 0x3B31AA86(993110662)

!--- !--- Output is truncated !---
```

Verify Whether the Lifetimes are Properly Configured

Use these commands to verify the current SA lifetime and the time for next renegotiation:

- **show crypto isakmp sa detail**
- **show crypto ipsec sa peer<NBMA-address-peer>**

Notice SA lifetime values. If they are close to the configured lifetimes (default is 24 hrs for ISAKMP and 1 hour for IPsec), then that means these SAs have been recently negotiated. If you look a little while later and they have been negotiated again, then the ISAKMP and/or IPsec can be bouncing up and down.

<#root>

```
Router#

show crypto ipsec security-assoc lifetime

Security association lifetime: 4608000 kilobytes/3600 seconds


Router#

show crypto isakmp policy

Global IKE policy
Protection suite of priority 1
Encryption algorithm: DES-Data Encryption Standard (65 bit keys)
Hash algorithm: Message Digest 5
Authentication method: Pre-Shared Key
Diffie-Hellman group: #1 (768 bit)

Lifetime: 86400 seconds, no volume limit

Default protection suite
 Encryption algorithm: DES- Data Encryption Standard (56 bit keys)
 Hash algorithm: Secure Hash Standard
 Authentication method: Rivest-Shamir-Adleman Signature
 Diffie-Hellman group: #1 (768 bit)
 Lifetime: 86400 seconds, no volume limit

Router#

show crypto ipsec sa

interface: Ethernet0/3
    Crypto map tag: vpn, local addr. 172.17.0.1
   local  ident (addr/mask/prot/port): (172.16.1.1/255.255.255.255/47/0)
   remote ident (addr/mask/prot/port): (172.17.0.1/255.255.255.255/47/0)
   current_peer: 172.17.0.1:500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 19, #pkts encrypt: 19, #pkts digest 19
    #pkts decaps: 19, #pkts decrypt: 19, #pkts verify 19
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0, #pkts decompress failed: 0
    #send errors 1, #recv errors 0
     local crypto endpt.: 172.16.1.1, remote crypto endpt.: 172.17.0.1
     path mtu 1500, media mtu 1500
     current outbound spi: 8E1CB77A

 inbound esp sas:
      spi: 0x4579753B(1165587771)
        transform: esp-3des esp-md5-hmac ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 2000, flow_id: 1, crypto map: vpn


sa timing: remaining key lifetime (k/sec): (4456885/3531)

        IV size: 8 bytes
        replay detection support: Y
outbound esp sas:
      spi: 0x8E1CB77A(2384246650)
        transform: esp-3des esp-md5-hmac ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 2001, flow_id: 2, crypto map: vpn


sa timing: remaining key lifetime (k/sec): (4456885/3531)

        IV size: 8 bytes
        replay detection support: Y
```

Verify Whether the Traffic Flows in Only One Direction

The VPN tunnel between the spoke-to-spoke router is up, but unable to pass data traffic.

<#root>

```
Spoke1#

show crypto ipsec sa peer 172.16.2.11

   local  ident (addr/mask/prot/port): (172.16.1.1/255.255.255.255/47/0)
   remote ident (addr/mask/prot/port): (172.16.2.11/255.255.255.255/47/0)


#pkts encaps: 110, #pkts encrypt: 110
    #pkts decaps: 0, #pkts decrypt: 0,

local crypto endpt.: 172.16.1.1,
remote crypto endpt.: 172.16.2.11
      inbound esp sas:
      spi: 0x4C36F4AF(1278669999)
      outbound esp sas:
      spi: 0x6AC801F4(1791492596)

!--- !--- Output is truncated !---


Spoke2#

sh crypto ipsec sa peer 172.16.1.1
```

```
   local  ident (addr/mask/prot/port): (172.16.2.11/255.255.255.255/47/0)
   remote ident (addr/mask/prot/port): (172.16.1.1/255.255.255.255/47/0)


 #pkts encaps: 116, #pkts encrypt: 116,
   #pkts decaps: 110, #pkts decrypt: 110,

local crypto endpt.: 172.16.2.11,
remote crypto endpt.: 172.16.1.1
     inbound esp sas:
     spi: 0x6AC801F4(1791492596)
     outbound esp sas:
     spi: 0x4C36F4AF(1278669999

!--- !--- Output is truncated !---
```

There is no decap packets in spoke1, which means esp packets are dropped somewhere in the path return from spoke2 towards spoke1.

The spoke2 router shows both encap and decap, which means that ESP traffic is filtered before it reaches spoke2. It can happen at the ISP end at spoke2 or at any firewall in path between spoke2 router and spoke1 router. After they allow ESP (IP Protocol 50), spoke1 and spoke2 both show encaps and decaps counters increment.

<#root>

```
spoke1#

show crypto ipsec sa peer 172.16.2.11

   local  ident (addr/mask/prot/port): (172.16.1.1/255.255.255.255/47/0)
   remote ident (addr/mask/prot/port): (172.16.2.11/255.255.255.255/47/0)


 #pkts encaps: 300, #pkts encrypt: 300
    #pkts decaps: 200, #pkts decrypt: 200


!--- !--- Output is truncated !---

spoke2#

sh crypto ipsec sa peer 172.16.1.1

   local  ident (addr/mask/prot/port): (172.16.2.11/255.255.255.255/47/0)
   remote ident (addr/mask/prot/port): (172.16.1.1/255.255.255.255/47/0)


#pkts encaps: 316, #pkts encrypt: 316,
   #pkts decaps: 300, #pkts decrypt: 310


!--- !--- Output is truncated !---
```

Verify that the Routing Protocol Neighbor is Established

Spokes are unable to establish routing protocol neighbor relationship:

<#root>

```
Hub#

show ip eigrp neighbors

H   Address     Interface   Hold Uptime     SRTT    RTO       Q      Seq
                                  (sec)             (ms)  Cnt     Num
2   10.0.0.9     Tu0         13  00:00:37     1    5000       1      0
0   10.0.0.5     Tu0         11     00:00:47   1587   5000       0     1483
1   10.0.0.11    Tu0         13     00:00:56     1    5000       1      0

Syslog message:
%DUAL-5-NBRCHANGE: IP-EIGRP(0) 10:
Neighbor 10.0.0.9 (Tunnel0) is down: retry limit exceeded


Hub#

show ip route eigrp

    172.17.0.0/24 is subnetted, 1 subnets
C       172.17.0.0 is directly connected, FastEthernet0/0
     10.0.0.0/24 is subnetted, 1 subnets
C       10.0.0.0 is directly connected, Tunnel0
C    192.168.0.0/24 is directly connected, FastEthernet0/1
S*   0.0.0.0/0 [1/0] via 172.17.0.100
```

Verify if NHRP multicast mapping is configured properly in the hub.

In the hub, it is required to have dynamic nhrp multicast mapping configured in the hub tunnel interface.

Configuration example:

```
interface Tunnel0
 ip address 10.0.0.1 255.255.255.0
 ip mtu 1400
 no ip next-hop-self eigrp 10
 ip nhrp authentication test
 ip nhrp network-id 10
 no ip split-horizon eigrp 10
 tunnel mode gre multipoint

!--- !--- Output is truncated !---
```

Configuration example with the correct entry for dynamic nhrp multicast mapping:

<#root>

```
interface Tunnel0
 ip address 10.0.0.1 255.255.255.0
 ip mtu 1400
 no ip next-hop-self eigrp 10
 ip nhrp authentication test


ip nhrp map multicast dynamic

 ip nhrp network-id 10
 no ip split-horizon eigrp 10
 tunnel mode gre multipoint

!--- !--- Output is truncated !---
```

This allows NHRP to automatically add spoke routers to the multicast NHRP mappings.

For more information, refer to the ip nhrp map multicast dynamic command in the Cisco IOS IP Addressing Services Command Reference.

<#root>

```
Hub#

show ip eigrp neighbors

IP-EIGRP neighbors for process 10
H   Address      Interface    Hold   Uptime    SRTT     RTO      Q      Seq
                                               (sec)    (ms)    Cnt     Num
2   10.0.0.9      Tu0          12   00:16:48    13      200      0      334
1   10.0.0.11     Tu0          13   00:17:10    11      200      0      258
0   10.0.0.5      Tu0          12   00:48:44   1017     5000     0      1495

Hub#

show ip route


     172.17.0.0/24 is subnetted, 1 subnets
C       172.17.0.0 is directly connected, FastEthernet0/0

D    192.168.11.0/24 [90/2944000] via 10.0.0.11, 00:16:12, Tunnel0

     10.0.0.0/24 is subnetted, 1 subnets
C       10.0.0.0 is directly connected, Tunnel0
C    192.168.0.0/24 is directly connected, FastEthernet0/1

D    192.168.2.0/24 [90/2818560] via 10.0.0.9, 00:15:45, Tunnel0

S*   0.0.0.0/0 [1/0] via 172.17.0.100
```

Routes to the spokes are learned through eigrp protocol.

Problem with Remote-access VPN with DMVPN Integration

Problem

DMVPN is works fine, but is unable to establish the RAVPN.

Solution

Use ISAKMP profiles and IPsec profiles to achieve this. Create separate profiles for the DMVPN and RAVPN.

For more information, refer to DMVPN and Easy VPN Server with ISAKMP Profiles Configuration Example.

Problem with Dual-hub-dual-dmvpn

Problem

Problem with dual-hub-dual-dmvpn. Specifically, tunnels go down and cannot re-negotiate.

Solution

Use the shared keyword in the tunnel IPsec protection for both the tunnel interfaces on the hub, and also on the spoke.

An configuration example:

```
interface Tunnel43
 description <<tunnel to primary cloud>>
 tunnel source interface vlan10
 tunnel protection IPSec profile myprofile shared

!--- !--- Output is truncated !---

interface Tunnel44
 description <<tunnel to secondary cloud>>
 tunnel source interface vlan10
 tunnel protection IPSec profile myprofile shared

!--- !--- Output is truncated !---
```

For more information, refer to the **tunnel protection** command in theCisco IOS Security Command Reference (A-C).

Trouble with Log on to a Server through DMVPN

Problem

Issue traffic through the DMVPN network server cannot be accessed.

Solution

The problem could be related to the MTU and MSS size of the packet which uses GRE and IPsec.

Now, the packet size could be an issue with the fragmentation. To eliminate this problem, use these commands:

<#root>

```
ip mtu 1400
ip tcp adjust-mss 1360
crypto IPSec fragmentation after-encryption (global)
```

You can also configure the **tunnel path-mtu-discovery**command to dynamically discover the MTU size.

For a more detailed explanation, refer toResolve IP Fragmentation, MTU, MSS, and PMTUD Issues with GRE and IPSEC.

Unable to Access the Servers on DMVPN Through Certain Ports

Problem

Unable to access servers on DMVPN through specific ports.

Solution

To verify disable the Cisco IOS firewall feature set and see if it works.

If it works fine, then the problem is related to the Cisco IOS firewall configuration, not with the DMVPN.

Related Information

- **Dynamic Multipoint VPN (DMVPN)**
- **IPSec Negotiation/IKE Protocols**
- **Cisco Technical Support & Downloads**