

vEdgeの双方向フォワーディング検出とデータプレーン接続の問題のトラブルシューティング

内容

[はじめに](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[コントロールプレーン情報](#)

[コントロールのローカルプロパティの確認](#)

[コントロール接続の確認](#)

[オーバーレイ管理プロトコル](#)

[OMP TLOCがvEdgeからアドバタイズされることを確認します。](#)

[vSmartがTLOCを受信してアドバタイズすることを確認する](#)

[双方向フォワーディング検出](#)

[show bfd sessionsコマンドについて](#)

[show tunnel statisticsコマンド](#)

[Access List](#)

[ネットワークアドレス変換](#)

[ツールstun-clientを使用してNATマップとフィルタを検出する方法](#)

[CLIで使用されるデータプレーントンネル「送信」用にサポートされるNATタイプ](#)

[ファイアウォール](#)

[セキュリティ](#)

[DSCPでマークされたトラフィックに関するISPの問題](#)

[BFDのデバッグ](#)

[関連情報](#)

はじめに

このドキュメントでは、コントロールプレーン接続後にサイト間でデータプレーン接続が確立されない場合のvEdgeデータプレーン接続の問題について説明します。

前提条件

要件

Ciscoでは、Cisco Software Defined Wide Area Network (SDWAN) ソリューションに関する知識を推奨しています。

使用するコンポーネント

このドキュメントの内容は、特定のソフトウェアやハードウェアのバージョンに限定されるものではありません。このドキュメントでは、vEdgeプラットフォームに重点を置いています。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、クリアな（デフォルト）設定で作業を開始しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

Ciscoエッジルータ(コントローラモードのCisco IOS® XEルータ)については、を参照してください。

コントロールプレーン情報

コントロールのローカルプロパティの確認

vEdgeのインターフ Wide Area Network (WAN) エイスのステータスを確認するには、コマンド `show control local-properties wan-interface-list` を使用します。

この出力では、RFC 4787を確認でき **Network Address Translation (NAT) Type** ます。

vEdgeがNATデバイス（ファイアウォール、ルータなど）の背後にある場合、パブリックおよびプライベートIPv4アドレス、パブリックおよびプライベート送信元 **User Datagram Protocol (UDP)** ポートを使用してデータプレーントンネルが構築されます。

また、トンネルインターフェイスの状態、色、および設定されているコントロール接続の最大数も確認できます。

```
vEdge1# show control local-properties wan-interface-list NAT TYPE: E -- indicates End-point independent mapping A -- indicates Address-port dependent
```

このデータを使用して、データトンネルの構築方法と、（ルータの観点から）データトンネルの構築時に使用するポートについての特定の情報を特定できます。

コントロール接続の確認

データプレーントンネルを形成しない色に、オーバーレイ内のコントローラとのコントロール接続が確立されていることを確認することが重要です。

それ以外の場合、vEdgeは経由でvSmartに情報 **Transport Locator (TLOC)** 報を送信し **Overlay Management Protocol (OMP)** せん。


コマ `show control connections` ンドを使用して動作可能かどうかを確認し、状態 `connect` 態を探すことができます。

```
vEdge1# show control connections PEER PEER CONTROLLER PEER PEER PEER SITE DOMAIN PEER PRIV PEER PUB GROUP TYPE PROT SY
```

データトンネルを形成しないインターフェイスが接続を試みる場合は、その色でコントロール接続を正常に起動して問題を解決します。

または、トンネルインターフェイスセクション `max-control-connections 0` で、選択したインターフェイスのを設定します。

```
vpn 0 interface ge0/1 ip address 10.20.67.10/24 tunnel-interface encapsulation ipsec color mpls restrict max-control-connections 0 no allow-service bgp all
```

 注：同じ目標を達成するために、`no control-connections` マンドを使用できる場合があります。ただし、このコマンドでは、コントロール接続の最大数は確立されません。このコマンドはバージョン15.4から廃止され、新しいソフトウェアでは使用されません。

オーバーレイ管理プロトコル

OMP TLOCがvEdgeからアドバタイズされることを確認します。

OMP TLOCを送信できません。インターフェイスがその色を介して制御接続を形成しようとし、コントローラに到達できないためです。

色 (データがトンネリングする色) がその特定の色のTLOCをvSmartsに送信するかどうかを確認します。


OMPピア `show omp tlocs advertised` に送信されるTLOCを確認するには、コマンドを使用します。

例：色 `mpls` と `gold` とTLOCはカラーmplsのvSmartに送信されません。

```
vEdge1# show omp tlocs advertised C -> chosen I -> installed Red -> redistributed Rej -> rejected L -> looped R -> resolved S -> stale Ext -> extranet Stg
```

例：色 `mpls` と `gold` と両方の色に対してTLOCが送信されます。

```
vEdge2# show omp tlocs advertised C -> chosen I -> installed Red -> redistributed Rej -> rejected L -> looped R -> resolved S -> stale Ext -> extranet Stg
```

 注：ローカルに生成されたコントロールプレーン情報の場合、「FROM PEER」フィールドは0.0.0.0に設定されます。ローカルで発信された情報を検索する場合は、この値に基づいて一致していることを確認します。

vSmartがTLOCを受信してアドバタイズすることを確認する

TLOCがvSmartにアドバタイズされます。正しいピアからTLOCを受信し、他のvEdgeにアドバタイズすることを確認します。

例：vSmartは10.1.0.2 vEdge1からTLOCを受信します。

```
<#root>
```

```
vSmart1# show omp tlocs received
```

```
C -> chosen I -> installed
```

```
Red -> redistributed Rej -> rejected L -> looped
```

R -> resolved

S -> stale Ext -> extranet Stg -> staged Inv -> invalid PUBLIC PRIVATE ADDRESS PSEUDO PUBLIC PRIVATE P
10.1.0.2 mpls ipsec 10.1.0.2 C,I,R 1 10.20.67.20 12386 10.20.67.20 12386 :: 0 :: 0 -

10.1.0.2 blue ipsec 10.1.0.2 C,I,R 1 198.51.100.187 12406 10.19.146.2 12406 :: 0 :: 0 -

10.1.0.30 mpls ipsec 10.1.0.30 C,I,R 1 10.20.67.30 12346 10.20.67.30 12346 :: 0 :: 0 - 10.1.0.30 gold

TLOCが表示されない場合、または他のコードが表示される場合は、次の点を確認してください。

<#root>

vSmart-vIPtela-MEX# show omp tlocs received

C -> chosen

I -> installed

Red -> redistributed

Rej -> rejected

L -> looped

R -> resolved

S -> stale Ext -> extranet Stg -> staged

Inv -> invalid

PUBLIC PRIVATE ADDRESS PSEUDO PUBLIC PRIVATE PUBLIC IPV6 PRIVATE IPV6 BFD FAMILY TLOC IP COLOR ENCAP F
10.1.0.2 mpls ipsec 10.1.0.2 C,I,R 1 10.20.67.20 12386 10.20.67.20 12386 :: 0 :: 0 -

10.1.0.2 blue ipsec 10.1.0.2 Rej,R,Inv 1 198.51.100.187 12406 10.19.146.2 12406 :: 0 :: 0 -

10.1.0.30 mpls ipsec 10.1.0.30 C,I,R 1 10.20.67.30 12346 10.20.67.30 12346 :: 0 :: 0 - 10.1.0.30 gold

TLOCをブロックするポリシーがないことを確認します。

show run policy control-policy - vSmartで、またはTLOCを拒否するtloc-listadvertised を探 received します。

<#root>

vSmart1(config-policy)# sh config policy lists tloc-list SITE20

tloc 10.1.0.2 color blue encap ipsec

!! control-policy SDWAN

sequence 10 match tloc tloc-list SITE20 ! action reject ---->

```
here we are rejecting the TLOC 10.1.0.2,blue,ipsec !! default-action accept !
apply-policy
site-list SITE20
```

```
control-policy SDWAN in ----->
```

the policy is applied to control traffic coming IN the vSmart, it will filter the tlocs before adding i



注:TLOCがまたはの場合、 **Rejected Invalid**他のvEdgeにアドバタイズされません。

TLOCがvSmartからアドバタイズされる時に、ポリシーによってTLOCがフィルタリングされないことを確認します。TLOCがvSmartで受信されますが、他のvEdgeでは受信されないことがわかります。

例1:C、I、RのTLOCを使用するvSmart

```
<#root>
```

```
vSmart1# show omp tlocs
```

```
C -> chosen I -> installed
```

```
Red -> redistributed Rej -> rejected L -> looped
```

```
R -> resolved
```

```
S -> stale Ext -> extranet Stg -> staged Inv -> invalid PUBLIC PRIVATE ADDRESS PSEUDO PUBLIC PRIVATE P
```

```
10.1.0.2 mpls ipsec 10.1.0.2 C,I,R 1 10.20.67.20 12386 10.20.67.20 12386 :: 0 :: 0 - 10.1.0.2 blue ipsec
```

```
10.1.0.30 mpls ipsec 10.1.0.30 C,I,R 1 10.20.67.30 12346 10.20.67.30 12346 :: 0 :: 0 - 10.1.0.30 gold
```

例2:vEdge1は、vEdge2のTLOCを青色から見ることはできません。MPLS TLOCのみを認識します。

```
<#root>
```

```
vEdge1# show omp tlocs C -> chosen I -> installed Red -> redistributed Rej -> rejected L -> looped R -> resolved S -> stale Ext -> extranet Stg -> staged I
```

```
10.1.0.2 mpls ipsec 10.1.0.3 C,I,R 1 10.20.67.20 12386 10.20.67.20 12386 :: 0 :: 0 up
```

```
10.1.0.30 mpls ipsec 10.1.0.3 C,I,R 1 10.20.67.30 12346 10.20.67.30 12346 :: 0 :: 0 up 10.1.0.30 gold
```

ポリシーを確認すると、TLOCがvEdge1に表示されない理由がわかります。

```
<#root>
```

```
vSmart1# show running-config policy policy lists tloc-list SITE20
```

```
tloc 10.1.0.2 color blue encap ipsec
```

```
! site-list SITE10 site-id 10 !! control-policy SDWAN sequence 10 match tloc
```

```
tloc-list SITE20
```

```
! action reject !! default-action accept !
apply-policy
site-list SITE10

control-policy SDWAN out
```

```
!
!
```

双方向フォワーディング検出

show bfd sessionsコマンドについて

出力で確認すべき主な項目を次に示します。

<#root>

```
vEdge-2# show bfd sessions SOURCE TLOC REMOTE TLOC DST PUBLIC DST PUBLIC DETECT TX SYSTEM IP SITE ID STATE COLOR COLO
10.1.0.5 10 down blue gold 10.19.146.2 203.0.113.225 4501 ipsec 7 1000 NA 7
10.1.0.30 30 up blue gold 10.19.146.2 192.0.2.129 12386 ipsec 7 1000 0:00:00:22 2 10.1.0.4 40 up blue
10.1.0.4 40 up mpls mpls 10.20.67.10
```


- **SYSTEM IP** : ピア system-ip
- **SOURCE and REMOTE TLOC COLOR** : これは、TLOCの送受信が予想される内容を知るために役立ちます。
- **SOURCE IP** : 送信 private 元IPです。NATの背後にいる場合、この情報がここに表示されます(を使用して確認でき show control local-properties <wan-interface-list ます)。
- **DST PUBLIC IP** : これは、NATの背後にあるかどうかにかかわらず、vEdgeがトン Data Plane ネルを形成するために使用する宛先です(例 : インターネットに直接接続されたvEdge、または Multi-Protocol Label Switching (MPLS) ンク)。
- **DST PUBLIC PORT** リモートvEdgeへのトン Data Plane ネルを形成するためにvEdgeが使用するパブリックNAT変換ポート。
- **TRANSITIONS:BFDセッションがステータスを変更した回数(間 NA で UP はその逆)。**

show tunnel statistics コマンド

は、データプレーントンネルに関する情報を表示で show tunnel statistics きます。vEdge間の特定のIPSECトンネルの packets を送受信するかどうかを決定できます。

これにより、パケットが両端に到着するかどうかを理解し、ノード間の接続の問題を切り分けることができます。

この例では、このコマンドを複数回実行すると、または増分が表示されるか、増分が表示されないこと tx-pkts とに注意でき rx-pkts ます。

 ヒント: tx-pkts のカウンタが増加している場合は、ピアにデータを送信します。rx-pkts が増加しない場合は、データがピアから受信されていないことを意味します。この場合は、もう一方の端をチェックして、tx-pkts が増加するかどうかを確認します。

<#root>

TCP vEdge2# show tunnel statistics

```
TUNNEL SOURCE DEST TUNNEL MSS PROTOCOL SOURCE IP DEST IP PORT PORT SYSTEM IP LOCAL COLOR REMOTE COLOR MTU tx-
ipsec 172.16.16.147 10.88.244.181 12386 12406 10.1.0.5 public-internet default 1441 38282 5904968 38276
ipsec 172.16.16.147 10.152.201.104 12386 63364 10.1.0.0 public-internet default 1441 33421 5158814 334
```

TUNNEL PROTOCOL	SOURCE IP	DEST IP	SOURCE PORT	DEST PORT	SYSTEM IP	LOCAL COLOR	REMOTE COLOR	MTU
ipsec	172.16.16.147	10.88.244.181	12386	12406	10.1.0.5	public-internet	default	38276
ipsec	172.16.16.147	10.152.201.104	12386	63364	10.1.0.0	public-internet	default	33421
ipsec	172.16.16.147	10.152.204.31	12386	58851	10.1.0.7	public-internet	public-internet	38276
ipsec	172.24.90.129	10.88.244.181	12426	12406	10.1.0.5	biz-internet	default	38276
ipsec	172.24.90.129	10.152.201.104	12426	63364	10.1.0.0	biz-internet	default	33421
ipsec	172.24.90.129	10.152.204.31	12426	58851	10.1.0.7	biz-internet	public-internet	38276

もう1つの便利なコマンド `show tunnel statistics bfd` は、特定のデータプレーントンネル内で送受信されたBFDパケットの数を確認するために使用できます。

```
vEdge1# show tunnel statistics bfd BFD BFD BFD BFD BFD BFD BFD PMTU PMTU PMTU PMTU TUNNEL SOURCE DEST ECHO TX ECHO RX BFD
```

Access List

出力を確認した後は、アクセスリストは便利に必要な `show bfd sessions` テップです。

プライベートIPとパブリックIPおよびポートが判明し `Access Control List (ACL)` たので、`SRC_PORT`、`DST_PORT`、`SRC_IP`、`DST_IP`と照合するACLを作成できます。

これは、送受信されたBFDメッセージの確認に役立ちます。

次に、ACL設定の例を示します。

```

policy access-list checkbfd-out sequence 10 match source-ip 192.168.0.92/32 destination-ip 198.51.100.187/32 source-port 12426 destination-port 12426 !
default-action accept
!
access-list checkbfd-in sequence 20 match source-ip 198.51.100.187/32 destination-ip 192.168.0.92/32 source-port 12426 destination-port 12426 ! action
vpn 0
interface ge0/0
access-list checkbfd-in in
access-list checkbfd-out out
!
!
!

```

この例では、このACLは2つのシーケンスを使用します。シーケンス10は、このvEdgeからピアに送信されるBFDメッセージと一致します。シーケンス20はその逆を行います。

送信元(Private)ポートおよび宛先(Public)ポートと照合されます。vEdgeがNATを使用している場合は、正しい送信元ポートと宛先ポートを確認してください。

各シーケンスカウンタのヒットを確認するには、 `show policy access-list counters <access-list name>`

```
vEdge1# show policy access-list-counters NAME COUNTER NAME PACKETS BYTES ----- checkbfd bfd-out-t
```

ネットワークアドレス変換

ツールstun-clientを使用してNATマップとフィルタを検出する方法

すべての手順を実行し、NATの背後にいる場合は、次の手順で動 **UDP NAT Traversal (RFC 4787) Map and Filter** 作を特定します。

このツールは、vEdgeがNATデバイスの背後にある場合に、ローカルvEdgeの外部IPアドレスを検出するために使用されます。

このコマンドは、デバイスのポートマッピングを取得し、オプションでローカルデバイスとサーバ間のNATに関するプロパティを検出します (パブリックサーバ : google stun serverなど)。



注 : 詳細については、[Docs Viptela - STUN Client](#)を参照してください。

<#root>

```

vEdge1# tools stun-client vpn 0 options "--mode full --localaddr 192.168.12.100 12386 --verbosity 2 stur
stunclient --mode full --localaddr 192.168.12.100 stun.l.google.com in VPN 0 Binding test: success
Local address: 192.168.12.100:12386
Mapped address: 203.0.113.225:4501
Behavior test: success

```


Nat behavior: Address Dependent Mapping

Filtering test: success

Nat filtering: Address and Port Dependent Filtering


新しいバージョンのソフトウェアでは、構文が少し異なる場合があります。

<#root>

```
vEdge1# tools stun-client vpn 0 options "--mode full --localaddr 192.168.12.100 --localport 12386 --ver
```

この例では、Google STUNサーバへのUDP送信元ポート12386を使用して、完全なNAT検出テストを実行します。

このコマンドの出力には、NATの動作とRFC 4787に基づくNATフィルタタイプが示されます。

 **注：**を使用するときは、トンネルインターフェイスでSTUNサービスを許可することを忘 `tools stun`れないでください。許可しないと、機能しません。STUNデータ `allow-service stun` を渡すために使用します。

<#root>

```
vEdge1# show running-config vpn 0 interface ge0/0 vpn 0 interface ge0/0 ip address 10.19.145.2/30 ! tunnel-interface encapsulation ipsec color gold max-
```

```
allow-service stun
```

```
! no shutdown !!
```

次に、STUN用語（フルコーンNAT）とRFC 4787（UDPのNAT動作）のマッピングを示します。

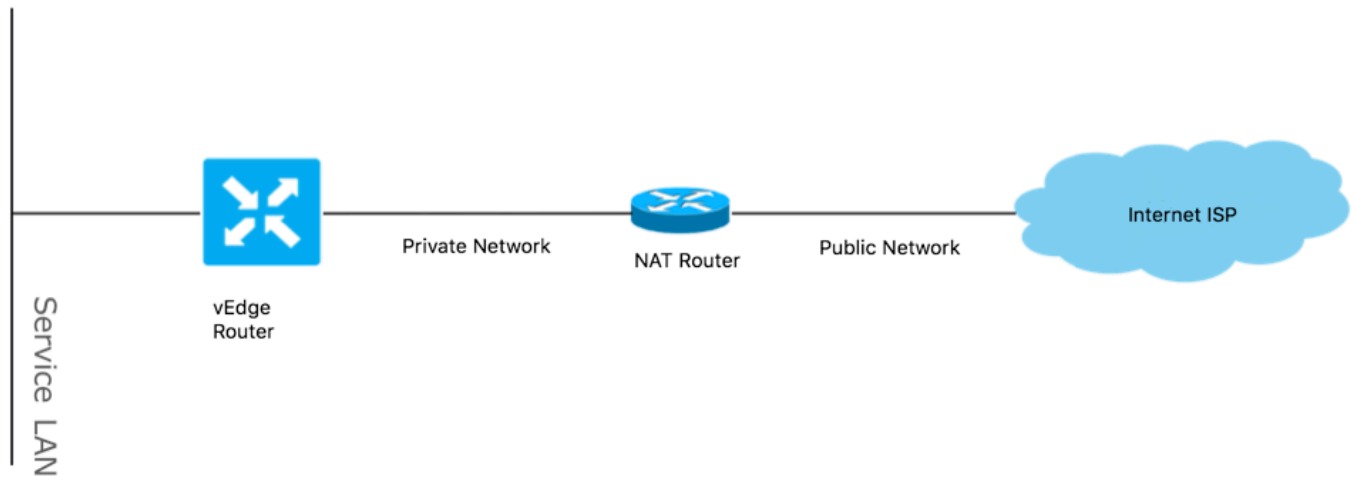
NAT Traversal Mapping Between used Viptela Terminologies		
STUN RFC 3489 Terminology	RFC 4787 Terminology	
	Mapping Behavior	Filtering Behavior
Full-cone NAT	Endpoint-Independent Mapping	Endpoint-Independent Filtering
Restricted Cone NAT	Endpoint-Independent Mapping	Address-Dependent Filtering
Port-Restricted Cone NAT	Endpoint-Independent Mapping	Address and Port-Dependent Filtering
Symmetric NAT	Address-and(or) Port-Dependent Mapping	Address-Dependent Filtering
		Address and Port-Dependent Filtering

CLIで使用されるデータプレーントンネル「送信」用にサポートされるNATタイプ

ほとんどの場合、ビジネスインターネットやパブリックインターネットなどのパブリックカラーをインターネットに直接接続できます。

また、vEdge WANインターフェイスと実際のインターネットサービスプロバイダー(ISP)の背後にNATデバイスが存在する場合があります。

このように、vEdgeはプライベートIPを持つことができ、他のデバイス(ルータ、ファイアウォールなど)はパブリック側のIPアドレスを持つデバイスにすることができます。



NATタイプが正しくない場合、データプレーントンネルの形成を許可しない最も一般的な理由の1つである可能性があります。サポートされているNATタイプは次のとおりです。

NAT Traversal Support		
Source	Destination	Supported (YES/NO)
Full-Cone NAT	Full-cone NAT	Yes
Full-Cone NAT	Restricted Cone NAT	Yes
Full-Cone NAT	Port-Restricted Cone NAT	Yes
Full-Cone NAT	Symmetric NAT	Yes
Restricted Cone NAT	Full-cone NAT	Yes
Restricted Cone NAT	Restricted Cone NAT	Yes
Restricted Cone NAT	Port-Restricted Cone NAT	Yes
Restricted Cone NAT	Symmetric NAT	Yes
Port-Restricted Cone NAT	Full-cone NAT	Yes
Port-Restricted Cone NAT	Restricted Cone NAT	Yes
Port-Restricted Cone NAT	Port-Restricted Cone NAT	Yes
Port-Restricted Cone NAT	Symmetric NAT	No
Symmetric NAT	Full-cone NAT	Yes
Symmetric NAT	Restricted Cone NAT	yes
Symmetric NAT	Port-Restricted Cone NAT	No
Symmetric NAT	Symmetric NAT	No

ファイアウォール

サポートされていない送信元および宛先タイプではなく、すでにNATをチェックしている場合は、トン Data Plane ネルの形成に使用されるポートがファイアウォールによってブロックされている可能性があります。

次のポートがファイアウォールでデータプレーン接続用を開いていることを確認します。 **vEdge to vEdge Data Plane:**

UDP 12346 ~ 13156

vEdgeからコントローラへのコントロール接続用 :

UDP 12346 ~ 13156

TCP 23456から24156

データプレーントンネルの接続を正常に行うために、これらのポートを開いていることを確認します。

データプレーントンネルに使用される送信元ポートと宛先ポートを確認する `show tunnel statistics` と、またはを使用 `show bfd sessions | tab` ですが、使用できません `show bfd sessions`。

送信元ポートは表示されず、宛先ポートだけが表示されます。

```
vEdge1# show bfd sessions SOURCE TLOC REMOTE TLOC DST PUBLIC DST PUBLIC DETECT TX SYSTEM IP SITE ID STATE COLOR COLOR
```



注 : 使用されるSD-WANファイアウォールポートの詳細については、[こちら](#)を参照してください。

セキュリティ

ACLカウンタのインバウンドとアウトバウンドが増加している場合は、いくつかの繰り返しを確認してください `show system statistics diff` and ensure there are no drops.

<#root>

```
vEdge1# show policy access-list-counters NAME COUNTER NAME PACKETS BYTES -----
```

```
checkbfd bfd-out-to-dc1-from-br1 55 9405
```

```
bfd-in-from-dc1-to-br1 54 8478
```

`rx_replay_integrity_drops` この出力では、 `show system statistics diff` command.

<#root>

```
vEdge1#show system statistics diff
```

```
rx_pkts : 5741427
```

```
ip_fwd : 5952166
```

```
ip_fwd_arp : 3
```

```
ip_fwd_to_egress : 2965437
```

```
ip_fwd_null_mcast_group : 26
```

```
ip_fwd_null_nhops : 86846
```

ip_fwd_to_cpu : 1413393
ip_fwd_from_cpu_non_local : 15
ip_fwd_rx_ipsec : 1586149
ip_fwd_mcast_pkts : 26
rx_bcast : 23957
rx_mcast : 304
rx_mcast_link_local : 240
rx_implicit_acl_drops : 12832
rx_ipsec_decap : 21
rx_spi_ipsec_drops : 16

rx_replay_integrity_drops : 1586035

port_disabled_rx : 2
rx_invalid_qtags : 212700
rx_non_ip_drops : 1038073
pko_wred_drops : 3
bfd_tx_record_changed : 23
rx_arp_non_local_drops : 19893
rx_arp_reqs : 294
rx_arp_replies : 34330
arp_add_fail : 263
tx_pkts : 4565384
tx_mcast : 34406
port_disabled_tx : 3
tx_ipsec_pkts : 1553753
tx_ipsec_encap : 1553753
tx_pre_ipsec_pkts : 1553753
tx_pre_ipsec_encap : 1553753
tx_arp_replies : 377
tx_arp_reqs : 34337
tx_arp_req_fail : 2
bfd_tx_pkts : 1553675
bfd_rx_pkts : 21
bfd_tx_octets : 264373160
bfd_rx_octets : 3600
bfd_pmtu_tx_pkts : 78
bfd_pmtu_tx_octets : 53052
rx_icmp_echo_requests : 48
rx_icmp_network_unreach : 75465
rx_icmp_other_types : 47
tx_icmp_echo_requests : 49655
tx_icmp_echo_replies : 48
tx_icmp_network_unreach : 86849
tx_icmp_other_types : 7
vEdge1# show system statistics diff

rx_pkts : 151
ip_fwd : 157
ip_fwd_to_egress : 75
ip_fwd_null_nhop : 3
ip_fwd_to_cpu : 43
ip_fwd_rx_ipsec : 41
rx_bcast : 1

rx_replay_integrity_drops : 41

rx_invalid_qtags : 7
rx_non_ip_drops : 21

```
rx_arp_non_local_drops : 2
tx_pkts : 114
tx_ipsec_pkts : 40
tx_ipsec_encap : 40
tx_pre_ipsec_pkts : 40
tx_pre_ipsec_encap : 40
tx_arp_reqs : 1
bfd_tx_pkts : 40
bfd_tx_octets : 6800
tx_icmp_echo_requests : 1
vEdge1# show system statistics diff
```

```
rx_pkts : 126
ip_fwd : 125
ip_fwd_to_egress : 58
ip_fwd_null_nhop : 3
ip_fwd_to_cpu : 33
ip_fwd_rx_ipsec : 36
rx_bcast : 1
rx_implicit_acl_drops : 1
```

rx_replay_integrity_drops : 35

```
rx_invalid_qtags : 6
rx_non_ip_drops : 22
rx_arp_replies : 1
tx_pkts : 97
tx_mcast : 1
tx_ipsec_pkts : 31
tx_ipsec_encap : 31
tx_pre_ipsec_pkts : 31
tx_pre_ipsec_encap : 31
bfd_tx_pkts : 32
bfd_tx_octets : 5442
rx_icmp_network_unreach : 3
tx_icmp_echo_requests : 1
tx_icmp_network_unreach : 3
```

vEdge1# show system statistics diff

```
rx_pkts : 82
ip_fwd : 89
ip_fwd_to_egress : 45
ip_fwd_null_nhop : 3
ip_fwd_to_cpu : 24
ip_fwd_rx_ipsec : 22
rx_bcast : 1
rx_implicit_acl_drops : 1
```

rx_replay_integrity_drops : 24

```
rx_invalid_qtags : 2
rx_non_ip_drops : 14
rx_arp_replies : 1
tx_pkts : 62
tx_mcast : 1
tx_ipsec_pkts : 24
tx_ipsec_encap : 24
tx_pre_ipsec_pkts : 24
tx_pre_ipsec_encap : 24
```

```
tx_arp_reqs : 1
bfd_tx_pkts : 23
bfd_tx_octets : 3908
rx_icmp_network_unreach : 3
tx_icmp_echo_requests : 1
tx_icmp_network_unreach : 3
vEdge1# show system statistics diff
```

```
rx_pkts : 80
ip_fwd : 84
ip_fwd_to_egress : 39
ip_fwd_to_cpu : 20
ip_fwd_rx_ipsec : 24
```

```
rx_replay_integrity_drops : 22
```

```
rx_invalid_qtags : 3
rx_non_ip_drops : 12
tx_pkts : 66
tx_ipsec_pkts : 21
tx_ipsec_encap : 21
tx_pre_ipsec_pkts : 21
tx_pre_ipsec_encap : 21
bfd_tx_pkts : 21
bfd_tx_octets : 3571
```

まず、vEdgeで**request security ipsec-rekey** を実行します。次に、の繰り返しを繰り返**show system statistics diff** し、まだ表示されているかどうかを確認し**rx_replay_integrity_drops**ます。

実行する場合は、セキュリティ設定を確認します。

```
vEdge1# show running-config security security
ipsec
authentication-type sha1-hmac ah-sha1-hmac
!
```

DSCPでマークされたトラフィックに関するISPの問題

デフォルトでは、vEdgeルータからコントローラへのすべての制御トラフィックと管理トラフィックは、DTLSまたはTLS接続を介して伝送され、DSCP値CS6 (10進数で48) でマークされます。

データプレーントンネルトラフィックの場合、vEdgeルータはIPsecまたはGREカプセル化を使用してデータトラフィックを相互に送信します。

データプレーン障害の検出とパフォーマンス測定のために、ルータは定期的に互いにBFDパケットを送信します。

これらのBFDパケットは、DSCP値CS6 (10進数で48) でもマーキングされます。

ISPの観点からは、このタイプのトラフィックはDSCP値がCS6のUDPトラフィックと同様に見なされます。これは、vEdgeルータとSD-WANコントローラが、デフォルトでマーキングされているDSCPを外部IPヘッダーにコピーするためです。

次に、tcpdumpが中継ISPルータで実行されている場合の例を示します。

```
14:27:15.993766 IP (tos 0xc0, ttl 64, id 44063, offset 0, flags [DF], proto UDP (17), length 168) 192.168.109.5.12366 > 192.168.20.2.12346: [udp sum ok]
```

ここに示すように、すべてのパケットはDSフィールドとも呼ばれるTOSバイト0xc0でマークされます (これは、10進数の192、つまり2進数の110 000 00に相当します)。

最初の6つの上位ビットは、10進数のDSCPビット値48またはCS6に対応します)。

出力の最初の2つのパケットはコントロールプレーントンネルに対応し、残りの2つのパケットはデータプレーントンネルトラフィックに対応します。

パケット長とTOSマークに基づいて、それがBFDパケット (RXおよびTX方向) であったことが確実に結論付けられます。これらのパケットもCS6でマークされます。

一部のサービスプロバイダー (特にMPLS L3 VPN/MPLS L2 VPNサービスプロバイダー) は、異なるSLAを維持し、DSCPマークに基づいて異なるクラスのトラフィックを異なる方法で処理できます。

たとえば、DSCP EFとCS6の音声およびシグナリングトラフィックに優先順位を付けるプレミアムサービスがある場合です。

アップリンクの合計帯域幅を超えていなくても、優先順位トラフィックはほぼ常にポリシングされるため、このタイプのトラフィックではパケット損失が発生し、BFDセッションもフラッピングする可能性があります。

サービスプロバイダールータ上の専用優先キューが枯渇していて、通常のトラフィック(vEdgeルータから単純なpingを実行する場合など)が廃棄されないことが判明した場合があります。

これは、このようなトラフィックがデフォルトのDSCP値0 (TOSバイト) でマーキングされるためです。

```
15:49:22.268044 IP (tos 0x0, ttl 62, id 0, offset 0, flags [DF], proto UDP (17), length 142) 192.168.110.5.12366 > 192.168.109.7.12346: [no cksum] UDP,
```

しかし同時に、BFDセッションはフラップします。

```
show bfd history DST PUBLIC DST PUBLIC RX TX SYSTEM IP SITE ID COLOR STATE IP PORT ENCAP TIME PKTS PKTS DEL -----
```

ここで、npingはトラブルシューティングに役立ちます。

```
vedge2# tools nping vpn 0 options "--tos 0x0c --icmp --icmp-type echo --delay 200ms -c 100 -q" 192.168.109.7 Nping in VPN 0 Starting Nping 0.6.47 (ht
```

BFDのデバッグ

より詳細な調査が必要な場合は、vEdgeルータでBFDのデバッグを実行します。

Forwarding Traffic Manager(FTM)はvEdgeルータ上のBFD操作を担当するため、必要 `debug ftm bfd`になります。

すべてのデバッグ出力はファイル `/var/log/tmplog/vdebug` に保存されており、これらのメッセージをコンソールに表示する場合は (Cisco IOSの `terminal monitor` 作と同様に) `monitor start /var/log/tmplog/vdebug`を使用でき `monitor start /var/log/tmplog/vdebug`ます。

ロギングを停止するには、 `monitor stop /var/log/tmplog/vdebug`

次に、タイムアウトによってダウンしたBFDセッションを出力で探す方法を示します(IPアドレス192.168.110.6のリモートTLOCには到達できません)。

```
log:local7.debug: May 7 16:23:09 vedge2 FTMD[674]: bfdmgr_session_update_state[1008]: BFD-session TNL 192.168.110.5:12366->192.168.110.6:123
```

有効にするもう一つの重要なデバッグは、 **Tunnel Traffic Manager (TTM) events** debug is `debug ttm events`です。

TTMの観点から見たイベント **BFD DOWN** の外観は次のとおりです。

```
log:local7.debug: May 7 16:58:19 vedge2 TTMD[683]: ttm_debug_announcement[194]: Received TTM Msg LINK_BFD, Client: ftmd, AF: LINK log:loc
```

関連情報

- [テクニカル サポートとドキュメント - Cisco Systems](#)

翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。