

仮想ルーティングおよび転送

- 機能の概要と変更履歴, on page 1
- 更新履歴, on page 1
- 機能説明 （2 ページ）
- VRF の設定 （4 ページ）
- VRF を使用した VLAN セグメンテーション （7 ページ）
- モニタリングおよびトラブルシューティング （9 ページ）

機能の概要と変更履歴

更新履歴

Table 1: マニュアルの変更履歴

改訂の詳細	リリース
UPF は、プライベート APN/DNN に対して最大 200 の VRF をサポートします。	2023.02.0
UPFは、プライベート APN/DNN に対して最大 129 の VRFをサポートします。	2022.04.0
次の機能のサポートが追加されました。 • 重複するIP プール • SMF と UPF 間の必須の VRF 順序の削除	2021.01.0
最初の導入。	2020.02.0

機能説明

仮想ルーティングおよびフォワーディング（VRF）は、同じルータ上で同時に複数のインスタンスのルーティングテーブルを共存させるためのテクノロジーです。ルーティングインスタンスは独立しているため、VRF は、互いに競合することなく、同じ IP アドレスまたは重複する IP アドレスを使用できます。

（注） 2023.02.0 以降のリリースでは、UPF はプライベート APN または DNN に対して最大 200 の VRF をサポートします。

2023.02.0 より前のリリースでは、UPF はプライベート APN または DNN に対して最大 129 の VRF をサポートしていました。

UPF では、この機能により、IP アドレスプールと VRF とのアソシエーションが可能になります。このプールからのチャンクは、これらのプールを使用する UPF に割り当てられます。UPF で VFR が関連付けられたプールは、Static または Private の種類のいずれかです。

UPF を登録する際、PRIVATE VRF プールのチャンクは、通常のプライベートプールと同様に割り当てられます。Static VRF プールの場合、SMF は、構成中に UPF へのチャンク割り当てを行います。UPF 登録時に事前に割り当てられた静的チャンクに対して Sx-Route-Update メッセージが送信されます。

重複するIPプール

重複プールは、IP アドレス範囲を共有して、使用します。重複プールのタイプは、[静的（Static）] または [プライベート（Private）] のいずれかです。パブリックプールを重複プールとして設定することはできません。各重複プールは、異なる VRF（ルーティングドメイン）およびプールグループに属します。また、1 つの APN が使用できるプールグループは 1 つのみであるため、重複プールは異なる APN に属します。

この機能がない場合、重複プールは SMF で構成されます。ただし、2 つの重複プールからのチャンクを同じ UPF に送信することはできません。つまり、UPF は 2 つの異なる重複プールのチャンクを処理できません。同じ IP アドレス範囲には、同じ数の UPF および重複プールが必要です。

この機能がある場合、UPF は 2 つの異なる重複プールのチャンクを処理できます。つまり、単一の UPF が、同じ IP 範囲を共有する重複プールをいくつでも処理できます。

同じ UPF 内の重複プールの機能は次のとおりです。

- 特定のプールのチャンクが UPF にインストールされると、対応する vrf-name がチャンクとともに送信されます。
- UPF は、VRF 対応のチャンクであり、チャンクを対応する VRF にインストールします。チャンクデータベースは、VRF でデータが格納されます。

- コールの割り当て、リリース、リカバリなど VPNMgr へのあらゆる通信には、UPF の対応する SessMgr に vrf-id が含まれます。これにより、VPNMgr は、指定された vrf-id でその IP の正しいチャンクを選択して処理できます。

UE IP VRF は、N4 SESSION ESTABLISHMENT 要求メッセージの VRF 名をカプセル化するカスタム IE です。

UE IP VRF 情報要素

プライベート UE IP VRF の IE フォーマットを次に示します。

表 *2 : UE IP VRF* フォーマット

	ビット							
オクテット	8	7	6	5	4	3	2	1
1 ～ 2	タイプ = 242（10 進数）							
3 ～ 4	長さ = n							
5	Spare					同一 VRF フラグ	IPv6 VRF が有効	IPv4 VRF が有効
m ～ m+1	VRF-1 名の長さ = p							
m+1 ～ m+1+p	VRF-1 名							
n ～ n+1	VRF-2 名の長さ = q							
m+1 ～ m+1+q	VRF-2 名							

次の表に、[UE IP VRF] フィールドで使用可能な値を示します。

ケース	UE IP VRF	値（2 進数）		
		ビット 3	ビット 2	ビット 1
1	IPv4 および IPv6 UE IP アドレスは、VRF に関連付けられません。	0	0	0
2	IPv4 UE IP アドレスのみが、VRF に関連付けられます。	0	0	1
3	IPv6 UE IP アドレスのみが、VRF に関連付けられます。	0	1	0
4	IPv4 と IPv6 の両方の UE IP アドレスが、異なる VRF に関連付けられます。	0	1	1

5	IPv4 と IPv6 の両方の UE IP アドレスが、共通の VRFに関連付けられます。	1	1	1

識別子としての VRF 名

VRF に関連する SMF と UPF 間の通信は、vrf-id 経由で行われていました。これには、オペレータが SMF と UPF の両方ですべての VRF を同じ順序で構成する必要がありました。

この機能では、vrf-name が VRF に関連する SMF と UPF 間のすべての通信で識別子として使用されます。この機能により、UPF のすべての VRF の構成が削除されます。オペレータは、SMF と UPF で異なる順序で VRF を構成でき、両方のノードで同じ vrf-name で VRF を識別できます。

制限事項と制約事項

次に、UPF の既知の制限事項および制約事項を示します。

- UPF は、VRF ベースの重複プールのみをサポートします。UPF は、NH ベースや VLAN ベースのプールなどの重複プールをサポートしません。
- UPF は、同じ UPF グループ内に複数の UPF を持つ静的 IP プールでの PDN Type IPv4v6 ベースのコールを許可しません。
- UPF は、VRF の動的更新をサポートしません。

VRF の設定

UPF で VRF サポートを構成するには、次の手順を実行します。

SMF で：

1. APN または DNN プロファイルを作成します。
2. 重複する IP プールを作成し、コンテキストレベルでそれぞれの APN、DNN と VRF を関連付けます。
3. APN または DNN を UPF プロファイルに関連付けます。

次に、SMF の構成例を示します：

```
profile dnn intershat1
.
.
.
   upf apn mpls1.com
exit
profile dnn intershat2
.
.
.
```

```
   upf apn mpls2.com
exit
profile network-element upf upf1
.
.
.
   dnn-list [ intershat1 intershat2 ]
exit
profile network-element upf upf2
.
.
.
   dnn-list [ intershat1 intershat2 ]
exit
ipam
   source local
   address-pool pool-intershat1
      vrf-name mpls-vrf-1@isp
      tags
         dnn intershat1
      exit
      ipv4
         address-range 209.165.201.25 255.255.255.224
      exit
   exit
   address-pool pool-intershat2
      vrf-name mpls-vrf-2@isp
      tags
         dnn intershat2
      exit
      ipv4
         address-range 209.165.201.25 255.255.255.224
      exit
   exit
exit
```

UPF で：

SMF からチャンクがプッシュされる前に、UPF で VRF を構成しておくことを推奨します。構成されていない場合、IP プールトランザクション全体（VRF に属さないチャンクを含む）でエラーが発生する場合があります。SMF は、しばらくしてから再試行します。

次に、UPF の構成例を示します。

UPF 1：

```
config
  context EPC2
    sx-service sx
      instance-type userplane
      bind ipv4-address 209.165.201.11 ipv6-address bbbb:aaaa::4
    exit
    user-plane-service up
      associate gtpu-service pgw-gtpu pgw-ingress
      associate gtpu-service sgw-ingress-gtpu sgw-ingress
      associate gtpu-service sgw-engress-gtpu sgw-egress
      associate gtpu-service saegw-sxu cp-tunnel
      associate sx-service sx
      associate fast-path service
      associate control-plane-group g1
    exit

  context isp
    ip vrf mpls-vrf-1
```

```
    #exit
    ip vrf mpls-vrf-2
    #exit
    ip vrf mpls-vrf-1
        route-distinguisher 61601 11100001
        route-target export 61601 11100001
        route-target import 61606 11100001
        route-target import 65200 11100001
    #exit
    address-family ipv4 vrf mpls-vrf-1
        redistribute connected
        redistribute static
    #exit
    address-family ipv6 vrf mpls-vrf-1
        redistribute connected
        redistribute static
    #exit
    ip vrf mpls-vrf-2
        route-distinguisher 61601 11100002
        route-target export 61601 11100002
        route-target import 61606 11100002
        route-target import 65200 11100002
    #exit
    address-family ipv4 vrf mpls-vrf-2
        redistribute connected
        redistribute static
    #exit
    address-family ipv6 vrf mpls-vrf-2
        redistribute connected
        redistribute static
    #exit
    #exit
apn mpls1.com
      pdp-type ipv4 ipv6
      bearer-control-mode mixed
      selection-mode sent-by-ms
      ip context-name isp
    exit
exit
control-plane-group g1
    peer-node-id ipv4-address 209.165.201.15
  #exit
  user-plane-group default
```

UPF 2：

```
config
  context EPC2
    sx-service sx
      instance-type userplane
      bind ipv4-address 209.165.201.12 ipv6-address bbbb:aaaa::5
    exit
    user-plane-service up
      associate gtpu-service pgw-gtpu pgw-ingress
      associate gtpu-service sgw-ingress-gtpu sgw-ingress
      associate gtpu-service sgw-engress-gtpu sgw-egress
      associate gtpu-service saegw-sxu cp-tunnel
      associate sx-service sx
      associate fast-path service
      associate control-plane-group g1
    exit
exit

  context isp
    ip vrf mpls-vrf-1
```

```
    #exit
    ip vrf mpls-vrf-2
    #exit
    apn mpls2.com
      pdp-type ipv4 ipv6
      bearer-control-mode mixed
      selection-mode sent-by-ms
      ip context-name isp
    exit
exit

control-plane-group g1
    peer-node-id ipv4-address 209.165.201.15
  #exit
  user-plane-group default
```

VRF を使用した VLAN セグメンテーション

表 *3 :* 機能の履歴

機能名	リリース情報	説明
仮想ルーティングと転送を使用した通信の VLAN セグメンテーションを有効にする	2024.01	UPF では、VRF を使用した VLAN セグメンテーションが許可されています。VRF は、確認します。 VLANに対して構成され、各VFRには、アップリンクパケットが送信されるデフォルトルートがあります。 **デフォルト設定** 無効：有効にするには構成が必要です。

機能説明

UPFは、仮想LAN（VLAN）を使用した企業通信のセグメント化をサポートしています。VLANは、1 つの物理 LAN セグメントをさらに小さな仮想ネットワークに分割し、ポートのグループ同士を、物理的に異なるセグメント上にあるかのように分離できるようにします。VLANセグメンテーションでは、異なるクラスに属するUEは、通信分離のために異なるVLANに割り当てられます。

VLAN セグメンテーションは、次の 2 つのメソッドを使用して実現できます。

1. Next Hop Forwarding
2. 仮想ルーティングおよび転送

UPFは、VRFを構成することにより、VLAN通信のセグメント化をサポートします。これは、VLAN ごとにデフォルトルートを使用して VRF を設定することで構成されるレイヤ 3 仮想化です。VRF デフォルトルートは、パケットの送信に使用されます。

VLAN セグメンテーションは、VLAN タグを SMF の IP プールに論理的にマッピングすることによって有効になります。UPF には、VRF に構成されたデフォルトルート経由で N6 インターフェイスでアップリンクパケットを転送する UPF に基づくマッピング構成が通知されます。

VRF を使用して VLAN セグメンテーションを有効にする

UPF で VRF を使用して VLAN セグメンテーションを有効にするには、次の構成を使用します。

```
config
  port ethernet slot/port
     no shutdown
     vlan vlan_tag_ID
     no shutdown
     bind interface interface_name context_name
     end
     context context_name
        ip vrf vrf_name
        { ip | ipv6 } address address subnetmask [ secondary ]
        end
```

注：

- **ip vrf** *vrf-name*：ルータに VRF を作成してVRF-ID を割り当てます。

設定例

次に、VRF を使用して VLAN セグメンテーションを有効にするための構成例を示します：

```
config
    port ethernet 1/10
       no shutdown
       vlan 400
         no shutdown
         bind interface N6_interface1 ISP
       #exit
       vlan 401
         no shutdown
         bind interface N6_interface2 ISP
       #exit
 context ISP
    ip vrf vrf1
    #exit
    ip vrf vrf2
    #exit
 interface N6_interface1
      ip vrf forwarding vrf1
      ip address 209.165.200.225 209.165.200.254
      ip address 209.165.201.1 209.165.200.254 secondary
      ipv6 address 2001:DB8::1/32 secondary
 interface N6_interface2
      ip vrf forwarding vrf2
      ip address 209.165.202.129 209.165.200.254
      ip address 209.165.201.30 209.165.200.254 secondary
      ipv6 address 2001:DB8:1::1/32 secondary
```

モニタリングおよびトラブルシューティング

この項では、機能のモニタリングと障害対応に使用できるCLIコマンドについて説明します。

showコマンドと出力

この項では、この機能のサポートにおけるshowコマンドおよびコマンドの出力について説明します。

show ip chunks

この CLI コマンドの出力には、そのコンテキストのすべてのチャンクが表示されます。

重複する IP プール機能を使用すると、**show ip chunks vrf** *vrf_name* CLI コマンドの出力には、その VRF のチャンクのみが表示されます。

- chunk-id
- chunk-size
- vrf-name
- start-addr
- end-addr
- used-addrs
- Peer Address

次に、出力例を示します。

```
==========================================
VRF Name: MPN00001
==========================================================================================================
|------------------------------|----------|------------|--------------|--------------|------------|
|          Peer Address          | chunk-id | chunk-size |   start-addr   |    end-addr
  | used-addrs |
|------------------------------|----------|------------|--------------|--------------|------------|
|                  192.10.25.23|1074790405|        8192|    36.40.128.0|  36.40.159.255|
           0|
|                  192.10.25.23|1074790406|        8192|    36.40.160.0|  36.40.191.255|
           0|
|------------------------------|----------|------------|--------------|--------------|------------|
```

show ipv6 chunks

この CLI コマンドの出力には、そのコンテキストのすべてのチャンクが表示されます。

重複する IP プール機能を使用すると、**show ipv6 chunks vrf** *vrf_name* CLI コマンドの出力は、その VRF にチャンクを表示します。

- chunk-id
- chunk-size

- vrf-name
- start-prefix
- end-prefix
- used-prefixes
- Peer Address

show ip bgp vpnv4

この CLI コマンドの出力には、すべての VPN ルーティング情報が表示されます。

重複する IP プール機能を使用すると、**show ip bgp vpnv4 vrf** *vrf_name* CLI コマンドは、その VRF に情報を表示します。

次に、出力例を示します。

```
VPNv4 Routing Table:
BGP table version is 1, local router ID is 172.31.35.36
Status Codes: s suppressed, d damped, h history, * valid, > best, i - internal, S stale,
 m Multipath
Origin Codes: i - IGP, e - EGP, ? - incomplete

Network              Next Hop             Metric LocPrf Weight Path
*>   0.0.0.0/0            172.31.35.29         0             0      65200 ?
*>   2.2.2.101/32         172.31.35.29         0             0      65200 ?
*>   2.2.3.2/32           0.0.0.0              0             32768  ?
*>   36.40.0.0/19         0.0.0.0              0             32768  ?
*>   36.40.32.0/19        0.0.0.0              0             32768  ?
*>   36.40.192.0/19       0.0.0.0              0             32768  ?

Total number of prefixes 6
```

show ip bgp vpnv6

この CLI コマンドの出力には、すべての VPN ルーティング情報が表示されます。

重複する IP プール機能を使用すると、**show ip bgp vpnv6 vrf** *vrf_name* CLI コマンドは、その VRF に情報を表示します。

次に、出力例を示します。

```
VPNv6 Routing Table:
BGP table version is 1, local router ID is 172.31.35.36
Status Codes: s suppressed, d damped, h history, * valid, > best, i - internal, S stale,
 m Multipath
Origin Codes: i - IGP, e - EGP, ? - incomplete

 Network              Next Hop             Metric LocPrf Weight Path
*>   0.0.0.0/0            172.31.35.29         0             0      65200 ?
*>   2036:adb0:40         172.31.35.29/51         0             32768  ?
*>   2036:adb0:40:2000    172.31.35.29/51           0             32768  ?
*>   2036:adb0:40:c000    172.31.35.29/51            0             32768  ?

Total number of prefixes 4
```

show mpls ilm

この CLI コマンドの出力には、FEC 情報を含む MPLS ILM テーブルが表示されます。

重複するIP プール機能を使用すると、**show mpls ilm** *fec verbose* and **show mpls ilm** *fec summary* CLI コマンドは、その VRF に情報を表示します。

次に、カウントを表示する **show mpls ilm** *fec summary* CLI コマンドの出力例を示します。

```
Total ILM entries: 406
```

次に、詳細な MPLS ILM テーブルを表示する **show mpls ilm** *fec Verbose* CLI コマンドの出力例を示します。

```
In-segment entry with in label: 832, id: 99, in label-space: 0, row status: Active
Owner: BGP, # of pops: 1 XC Index:708
FEC: 36.40.0.0/19
FEC: 36.40.32.0/19
FEC: 36.40.192.0/19

In-segment entry with in label: 833, id: 100, in label-space: 0, row status: Active
Owner: BGP, # of pops: 1 XC Index:709
FEC: 2037:adb0:201::/51
FEC: 2037:adb0:201::/51
FEC: 2037:adb0:201::/51
```

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。