



## VPP のサポート

---

Vector Packet Processing (VPPMOB) は、オープンソース ソリューションである fd.io の VPP に基づくモビリティ中心のソリューションです。特に IP 転送、回送、およびプロトコルの分野で [fd.io](https://www.fd.io/) の開発を活用しています。

- [マニュアルの変更履歴 \(2 ページ\)](#)
- [課金サポート \(2 ページ\)](#)
- [ルールベースによる遅延課金 \(2 ページ\)](#)
- [フローのアイドルタイムアウト \(3 ページ\)](#)
- [HTTP のサポート \(3 ページ\)](#)
- [IP 再アドレス指定 \(3 ページ\)](#)
- [DNS アドレス再指定先サーバーリスト \(4 ページ\)](#)
- [LTE ハンドオーバー \(6 ページ\)](#)
- [ネクストホップ \(6 ページ\)](#)
- [PDN の更新 \(6 ページ\)](#)
- [ポリシング \(6 ページ\)](#)
- [Pure-S のサポート \(8 ページ\)](#)
- [サービススキーマを介した応答ベースの課金 \(8 ページ\)](#)
- [サービススキーマを介した応答ベースの TRM \(8 ページ\)](#)
- [ToS マーキング \(9 ページ\)](#)
- [ボリュームベースのオフロード \(9 ページ\)](#)
- [サポートされる機能 \(9 ページ\)](#)
- [制限事項 \(10 ページ\)](#)
- [ユーザープレーンサービスでの高速パスの有効化 \(10 ページ\)](#)
- [SI プラットフォームでの VPP の有効化 \(11 ページ\)](#)
- [VPP 高速パスのモニタリングと障害対応 \(11 ページ\)](#)
- [VPP 設定パラメータのオーバーライドのサポート \(12 ページ\)](#)

## マニュアルの変更履歴



(注) リリース 21.24 よりも前に導入された機能については、詳細な改訂履歴は示していません。

改訂の詳細	リリース
このリリースでは、CUPS のフローごとの Fast Path 情報に対する VPP ctrl のサポートを追加。	21.27.x
このリリースでは、DNS アドレス再指定先サーバーリストのサポートを追加。	21.25.4
初版	21.24 より前

## 課金サポート

使用状況レポートは、コールの削除またはボリュームや時間のしきい値違反が発生すると、課金サーバーに通知されます。

ストリームがユーザープレーンで作成されると、課金を含むフローは、ストリーム作成中に設定された課金固有の操作に関連付けられます。オフロードと非オフロードの両方に関するすべてのフローの課金カウンタは、高速パスで維持されます。

ボリュームしきい値のオーバーフロー時に、高速パスはバケットカウンタを含む通知を送信し (PUSH モード)、時間しきい値に達した場合、アプリケーションは高速パスから課金カウンタを読み取ります (PULL モード)。ユーザープレーンは、これらのカウンタをそれぞれの URR で集約し、Sx インターフェイスを介して使用状況レポートをトリガーします。



**重要** このリリースでは、ボリュームと時間しきい値の両方で URR がサポートされています。複数の SDF と 1 つのベアラレベル URR がサポートされています。

## ルールベースによる遅延課金

サポートされる delay-charging のフレーバーは次のとおりです。

- Charge-to-application all-packets : フローのすべての制御パケット (ハンドシェイク、セッション中、およびティアダウン) が、アプリケーションパケットに一致する charging-action に基づいて課金されます。

- **Charge-to-application initial-packets** : フローのハンドシェイクパケットが、アプリケーションパケットに一致する **charging-action** に基づいて課金されます。
- **Charge-to-application tear-down-packets** : フローのティアダウンパケットが、アプリケーションパケットに一致する **charging-action** に基づいて課金されます。
- **Charge-separate-from-application** : すべての制御パケットのルール照合を行い、最も優先順位の高いルールに基づいて課金されます。

上記のすべてのシナリオで、遅延するのは課金のみですが、ルール照合はパケットの内容を対象に行われます。

**重要**

- **[Charge-separate-from-application mid session packets]** はサポートされません。オフロードされたフローは、引き続き最後に一致したルールに一致します。

遅延課金機能を有効にすると、TCP ハンドシェイクパケットは到着時にルールにヒットします。TCP ハンドシェイクパケットは、設定に基づく IP または TCP ルールにヒットします。**show active-charge** CLI コマンドを実行すると、TCP ハンドシェイクパケットがまだデフォルトルールにヒットしていることがわかります。最初の L7 パケットが到着するまでは、このルールによる課金は考慮されません。最初の L7 パケットが L7 ルールにヒットすると、クォータ要求の送信時に、L7 パケットと TCP ハンドシェイクパケットが同じ L7 RG に加えられます。

## フローのアイドルタイムアウト

設定可能な **idle-time out** がサポートされ、最大値は 24 時間です。以前のリリースでは、一部の特定の値に対してのみ使用がサポートされていました。

## HTTP のサポート

このリリースでは、HTTP トラフィックの分析と、このような HTTP ベースルールのポリシー照合がサポートされています。HTTP フローのオフロードは、**WebSocket**、**CONNECT** メソッド、または要求/応答内にコンテンツが存在する場合にのみサポートされます。

## IP 再アドレス指定

このリリースでは、IPv4 および IPv6 の IP 再アドレス指定がサポートされています。

IP 再アドレス指定は、課金アクションに関連付けられた課金ルールまたは後処理ルールを使用して設定できます。

ストリームは、IP 再アドレス指定動作セットとともにそれらのルールに一致するフローの高速パスで作成されます。一致するすべてのフロー（オフロードと非オフロードの両方）の高速パスに IPv4/IPv6 アドレスが設定されます。

## DNS アドレス再指定先サーバーリスト

許可されていない DNS サーバーを使用すると必ず、要求が変更され、DNS IP アドレスを再指定し、許可されたサーバーを使用するように求められます。**Ruledef** によって、パケットが DNS クエリに属しているかどうか、および DNS クエリが一連の許可された DNS サーバーに属しているかどうかを判別します。DNS クエリが許可された DNS サーバーに属していない場合、フローアクションが **readdress-server-list** から DNS サーバーを取得することになります。

**readdress-server-list** はアクティブ課金サーバーで設定されます。フローが **ruledef** に一致する場合に **readdress-server-list** のサーバーを使用するようにフローアクションを設定できます。

**readdress-server-list** で **active-charging service** を次のように設定します。

```
configure
  active-charging service service_name
    readdress-server-list name_of_list
      server ipv4_address [ port ]
      server ipv6_address [ port ]
```



(注) **readdress-server-list** には最大 10 のサーバーを設定でき、[active-charging service] には最大 10 の **readdress-server-lists** を設定できます。同じ **readdress-server-list** 内に、IPv4 アドレスと IPv6 アドレスの両方を設定できます。

次の 2 つの方法のいずれかを使用して、リストから **readdress-server-list** を選択します。

- ラウンドロビン：新しいフローごとにラウンドロビン方式でサーバーが選択されます。選択時に、リスト内の非アクティブサーバーは考慮されません。
- 階層型：このアプローチでは、サーバーは **readdress-server-list** で定義されている順序に応じて、プライマリ、セカンダリ、ターシャリというようにタグ付けされます。すべてのフローのアドレス再指定先は、プライマリサーバーが使用可能である限り、プライマリサーバーです。プライマリサーバーがダウンした場合は、フローのアドレス再指定先はセカンダリサーバーとなり、その後も同じロジックが繰り返されます。プライマリサーバーが再びアクティブになると、フローのアドレス再指定先はプライマリサーバーに戻ります。

次の CLI コマンドは、サーバー選択のアプローチを定義します。

```
charging-action action_name
  flow action readdress-server-list name_of_list [ hierarchy | round-robin ]
```

前述のコードに記載されている CLI コマンドでオプションが指定されていない場合、**round-robin** オプションがデフォルトのオプションと見なされます。

[active-charging service] で次の CLI コマンドを設定します。

```
configure
  active-charging service service_name
    readdress-server-list name_of_list
      server ipv4_address [ port ]
      server ipv6_address [ port ]
      consecutive-failures integer_value
      response-timeout integer_value
      reactivation-time integer_value

      charging-action action_name
        flow action readdress server-list name_of_list
    exit
```



(注) 前述のコードに記載されている CLI コマンドを設定するにあたり、次の値を検討してください。

- **consecutive-failures** : 整数値で、1 ~ 10 の範囲で指定する必要があります。デフォルト値は 5 です。
- **response-timeout** : 整数値で、1 ~ 10000 ミリ秒の範囲で指定する必要があります。デフォルト値は 1000 です。
- **reactivation-time** : 整数値で、1 ~ 1800 秒の範囲で指定する必要があります。デフォルト値は 300 です。

### アドレス再指定先サーバーの状態

アドレス再指定先サーバーの状態は、次のとおりです。

- アクティブ状態 : 一度設定すると、すべてのサーバーが [Active] としてマークされます。
- 非アクティブ状態 : アドレス再指定先サーバーからの応答が受信されない場合、サーバーは [Inactive] としてマークされます。
- アクティブ保留状態 : サーバーが [Active-Pending] 状態になると、アドレス再指定要求を受け付け可能となります。この状態のサーバー宛てに要求のアドレスが再指定され、応答が返ると、このサーバーの状態は [Active] に変更されます。それ以外の場合は、[Inactive] 状態に戻ります。

## LTE ハンドオーバー

次のタイプのハンドオーバーがサポートされています。

- X2 ベースのハンドオーバーの S-GW 再配置 (OI は 1 に設定)。
- S1 ベースのハンドオーバーの S-GW 再配置 (OI は 0 に設定)。
- eNodeB F-TEIDu の更新。

S-GW 再配置の場合、次の組み合わせがサポートされています。

- P-GW アンカーコール。
- Collapsed コールへの P-GW アンカーコール。
- P-GW アンカーコールへの Collapsed コール。

## ネクストホップ

このリリースでは、IPv4 および IPv6 のネクストホップアドレスがサポートされています。

ネクストホップアドレスは、課金アクションに関連付けられた課金ルールまたは後処理ルールを使用して設定できます。

ストリームは、これらのルールに一致するフローの高速パスでネクストホップ動作セットとともに作成されます。一致するすべてのフロー (オフロードと非オフロードの両方) の高速パスにネクストホップアドレスが設定されます。

## PDN の更新

PDN 更新手順は、このリリースの VPP でサポートされています。

Gx 手順を介してルールの追加、変更、削除が受信されるたびに、すべてのフローが SM-U にオンロードされます。これらのオンロードされたフロー上のすべてのパケットは、SM-U に送信されます。フローのトランスポート レベル マーキングおよび課金パラメータが変更された場合も、フローはオンロードされます。これらのフローは、ルール一致条件が変更されるか、トランザクションルール マッチング (TRM) が再び関与するパケットで再びオフロードされます。

## ポリシング

ポリサー設定では、セッションマネージャからの入力を使用します。これらの入力は、PCRF から AMBR として、またはフローレベルの QoS 情報から受信されます。セッションレベルの

AMBR ポリシングでは、PCRF から受信した値を常に受け付けます。ただし、フローレベルのポリシングが優先される場合（それが使用可能な場合）、AMBR ポリシングが順番に適用されます。つまり、ポリサーエンジンは階層型ポリシングを適用します。最初にフローレベル/ルールの帯域幅制限が適用され、次にセッションレベルの帯域幅制限が適用されます。



(注) RAR または CCA-U を介したセッション実行時の AMBR の変更が適用されます。

セッションマネージャから受信した入力値は、ポリサー設定およびポリサートークンバケットにプッシュされます。アップリンクまたはダウンリンクの各方向に対し、ポリサー設定とポリサートークンバケット用の新しいレコードが作成されます。

ポリサー設定はポリサーエンジンの参照先であり、ポリサートークンバケットは値の計算と復元に使用されます。

現在、ポリシングは、PCRF から受信した AMBR およびダイナミックルールのルールレベルの QoS 情報でサポートされています。静的ルールおよび事前定義ルールの場合、帯域幅制限は帯域幅ポリシー設定によって実施されます。コントロールプレーンの [Active Charging Service Configuration] モードの帯域幅ポリシー設定で設定された拡張ビットレートは、設定プッシュメカニズムの一部としてユーザープレーンに提供され、ユーザープレーンによるポリシングにも同じものが適用されます。以下に、帯域幅ポリシーの設定例を示します。

```
configure
  active-charging service ACS
    bandwidth-policy BWP

      flow limit-for-bandwidth id 1 group-id 1

      flow limit-for-bandwidth id 2 group-id 2
        group-id 1 direction uplink peak-data-rate 256000 peak-burst-size 32000
        violate-action discard
        group-id 1 direction downlink peak-data-rate 256000 peak-burst-size 32000
        violate-action discard
        group-id 2 direction uplink peak-data-rate 128000 peak-burst-size 16000
        violate-action discard
        group-id 2 direction downlink peak-data-rate 56000 peak-burst-size 7000
        violate-action discard
      exit
```

### 制限事項

このリリースでは、ポリシングに次の制限があります。

- **bandwidth-policy** の変更はサポートしません。
- ITC 帯域幅制限、トークンの補充（APN レベルおよび ACL レベルの両方）などの他の機能とのインタラクションはサポートされません。
- 現在、ポリサーベースの統計はサポートされていません。



- (注) 現在、ポリサー統計はサポートされていないため、オペレータはネットワークパフォーマンス モニタリング ツールを使用して帯域幅制限を確認できます。

## Pure-S のサポート

Pure-S デフォルトベアラー VPP 統合が CUPS アーキテクチャでサポートされるようになりました。これまで、CUPS での Pure-S コールは IFTASK を使用することでサポートされていました。今後は、Pure-S コールデータパスも VPP を使用します。

Pure-S コールの VPP 統合の一環として、SAEGW-UP のコールは、1 方向あたり 1 つのベアラー ストリーム (3 タプル: GTPU サービス IP アドレス、TEID、VRF ID) をインストールし、1 方向あたり 1 つの TEP 行も作成されます。

### サポートされる機能:

Pure-S でサポートされる機能は、次のとおりです。

- MME とネットワーク開始型シナリオ (MBR/CBR/UBR/DBR) 間のコリジョンに関するほとんどの手順。
- DBCmd および BRCmd コマンド。
- SAEGW-UP は、[IDLE] から [ACTIVE] に遷移中の IPv4 から IPv6 へ、または IPv6 から IPv4 への IP トランスポートの動作、および S1-u インターフェイスでのハンドオーバー手順をサポートします。接続時に S1-u で選択されたトランスポートもサポートされます。たとえば、IPv4 eNodeB から IPv6 eNodeB への eNode ハンドオーバーなどです。

## サービススキーマを介した応答ベースの課金

HTTP リクエストは、HTTP レスポンスの一致した課金アクションに対して課金されます。

## サービススキーマを介した応答ベースの TRM

アップリンクストリームのトランザクションルール マッチング (TRM) は、HTTP レスポンスを受信した後にのみ実行されます。



## ToS マーキング

### 機能説明

このリリースでは、IPv4 および IPv6 の ToS マーキングがサポートされています。

内部 IP ToS マーキングアドレスは、課金アクションに関連付けられた課金ルールまたは後処理ルールを使用して設定できます。外部 IP ToS マーキングは、コントロールプレーンで設定された QCI-DSCP マーキングテーブルを使用して実行されます。

ストリームは、これらのルールに一致するフローの高速パスで動作セットとともに作成されます。一致するすべてのフロー（オフロードと非オフロードの両方）には、高速パスで IPv4/IPv6 ToS マーキングが設定されます。

## ボリュームベースのオフロード

HTTP プロトコルの場合、要求/応答のコンテンツ（存在する場合）は、フロー内の各トランザクションの fastpath にオフロードされます。コンテンツの最後のパケットがストリームをパッシブ状態に戻し、パケットがセッションマネージャに到達します。

## サポートされる機能

このリリースでは、次のコールフレーバーがサポートされます。

- Pure-P IPv4/IPv6 コール
- Collapsed IPv4/IPv6 コール
- デフォルトベアラ
- Pure-S 機能
- 専用ベアラ
- ハンドオーバー

このリリースでは、次の機能がサポートされます。

- ペイロードパケット（課金アクション）および外部 GTP-U パケット（QCI/QoS マッピングテーブル）の ToS マーキング。
- ネクストホップ機能（IPv4/IPv6）。
- IP アドレス再設定機能（IPv4/IPv6）。
- アクションが [discard] の後処理ルール。
- アクションが [Next hop forwarding (IPv4/IPv6)] の後処理ルール。

- アクションが [ToS marking (UL and DL)] の後処理ルール。
- アクションが [Readdressing (IPv4/IPv6)] の後処理ルール。
- URR 機能 (Gz のみ) : SDF 1 つ、ベアラレベルの URR 1 つ。
- Gz 課金のみがサポートされます。
- VPP でのフラグメンテーションとリアセンブルがサポートされます。
- HTTP トラフィックポリシー照合がサポートされます。HTTP オフロードがサポートされるのは、CONNECT および WebSocket 要求のみです。
- このリリースは、サブスクリバあたり、すべてのアプリケーションを合わせて最大 5000 フローをサポートすることが検証されています。この制限はソフトウェアの性能によるものではありませんが、運用上推奨される制限です。この制限を超えるとアプリケーション障害が発生する可能性があるため、[Rulebase Configuration] モードで次の CLI を設定することを推奨します：**flow limit-across-applications 5000**

## 制限事項

次の機能は、このリリースではサポートされていません。

- Gy と Rf は個々にサポートされますが、同じサブスクリバに対して両方を同時に有効にすることはできません。
- Fast Path CLI が以前に有効化されている場合は、無効にできます。ただし、ユーザープレーンをリロードする必要があります。
- **VPPクラッシュログのサポート**：クラッシュレコードとミニコアファイルの生成がサポートされます。VPP の完全なコアファイルの生成はサポートされていません。

## ユーザープレーンサービスでの高速パスの有効化

ユーザープレーンサービスで Fast Path (VPP) を有効にするには、次の CLI コマンドを使用します。

```
configure
context context_name
  user-plane-service service_name
  associate fast-path service
end
```

注：

- **fast-path** : Fast Path 関連のパラメータを指定します。
- **service** : Fast Path 関連の設定を指定します。

## SI プラットフォームでの VPP の有効化

VPP を起動するには、次の手順を実行します。

1. ホストマシンにログオンし、staros\_param.cfg ファイルを含む ISO イメージを作成します。
2. FORWARDER\_TYPE=vpp という行を含むファイルを作成します。
3. 次のように、staros\_param.cfg ファイルを含む ISO ファイルを作成します。

```
genisoimage -l -o ssi_vpp.iso -r vppiso/
```

genisoimage がインストールされていない場合は、次のコマンドを実行します。

```
sudo apt-get install genisoimage
```

4. VM が実行されている場合は停止します。  

```
virsh destroy <vm_name>
```
5. VPP がフォワーダとして識別されていない VM にディスクがすでに接続されている場合は、そのディスクを切り離します。

VM で **dumpxml** コマンドを実行して、ディスクが接続されているか確認します。

ディスクを切り離すには、次のコマンドを実行します。

```
virsh detach-disk <vm_name> hdc -config
```

6. staros\_param.cfg ファイルを含む ISO ファイルを添付します。

```
virsh attach-disk <vm_name> <Path_of_ISO_FILE> hdc -type cdrom -config
```

## VPP 高速パスのモニタリングと障害対応

フローがオフロードされているか判断するには、次の CLI コマンドの出力で高速パスの統計情報を確認します。

- **show subscribers user-plane-only full all**
- **show user-plane-service all**
- **show user-plane-service statistics analyzer name ip**
- **show user-plane-service statistics analyzer name ipv6**
- **show user-plane-service statistics analyzer name tcp**
- **show user-plane-service statistics analyzer name udp**
- **show user-plane-service statistics analyzer name http**

フローごとの高速パス情報を確認するには、次の CLI コマンドの出力で Fast Path Info フィールドを確認します。

- **show subscribers user-plane-only callid call\_id flows full**

## VPP 設定パラメータのオーバーライドのサポート

VPP 設定パラメータの設定については、『*VPC-SI Administration Guide*』[英語]を参照してください。これらのパラメータはオーバーライドできます。オーバーライド値の特定については、シスコのアカウント担当者にお問い合わせください。

## 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。