

ASR5K の OCS 障害解決における障害処理と到達不能サーバ機能を設定する

内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[設定](#)

[ネットワーク図](#)

[Tx-Expiry](#)

[応答タイムアウト](#)

[Diameter セッション フェールオーバー](#)

[FH メカニズム](#)

[FH メカニズムの設定](#)

[FH メカニズムのデフォルトの動作](#)

[FH メカニズムの詳細なコールフロー](#)

[SU メカニズム](#)

[SU メカニズムの設定](#)

[SU メカニズムのコールフロー](#)

[FH および SU の設定例](#)

[確認](#)

[トラブルシュート](#)

[関連情報](#)

概要

このドキュメントでは、オンライン課金システム (OCS) で発生した問題の解決、またはポリシー/課金実施機能 (PCEF) と OCS 間の接続に関する問題の解決のために Gy インターフェイスで Failure-Handling (FH) と Server-Unreachable (SU) メカニズムを設定する方法について説明します。このドキュメントの情報は、Cisco 5000 シリーズ Aggregated Services Router (ASR5K) で稼働する Home Agent (HA)、Gateway General Packet Radio Service (GPRS) Support Node (GGSN)、および Packet Data Network Gateway (PGW) 機能に適用されます。

前提条件

要件

FH および SU メカニズムを使用するために、ご使用のシステムが次の要件を満たしていることを確認してください。

- Enhanced Charging Service (ECS) が使用可能である。
- HA、GGSN、または PGW 内に PCEF が存在している。
- Database を介した適切な Diameter 接続がある。
- Diameter Credit Control Application (DCCA) が使用可能である。

使用するコンポーネント

このドキュメントの情報は、ASR5K のすべてのバージョンに基づくものです。

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、初期 (デフォルト) 設定の状態から起動しています。対象のネットワークが実稼働中である場合には、どのようなコマンドについても、その潜在的な影響について確実に理解しておく必要があります。

背景説明

PCEF は Gy インターフェイス経由で OCS に接続され、Gy インターフェイスはベース プロトコルおよび DCCA として Diameter を使用します。次に PCEF と OCS 間のメッセージ フローを示します。

- **Credit Control Request (CCR)** : このメッセージは PCEF から OCS に送信されます。CCR メッセージには、Initial、Update、および Terminate の 3 種類があります。
- **Credit Control Answer (CCA)** : このメッセージは OCS から PCEF に対し、CCR のへの応答として送信されます。CCA メッセージにも、Initial、Update、および Terminate の 3 種類があります。
- **Re-Authorization Request (RAR)** : このメッセージは、セッションの再認証が必要な場合に OCS から PCEF に送信されます。

• **Re-Authorization Answer (RAA)** : これは、PCEF から OCS への RAR に対する応答です。このメッセージ フローを実現するため、PCEF と OCS 間で Diameter 接続が確立されます。OCS が否定的なメッセージを送信するか、PCEF と OCS 間でトランスポート接続が失敗するか、またはメッセージがタイムアウトになる可能性があります。これらの状況が原因で、サブスクライバセッションを確立できないことがあります。これにより、サブスクライバがサービスを使用できなくなることがあります。

この問題を解決するため、次の 2 つのメカニズムを使用できます。

- FH メカニズム

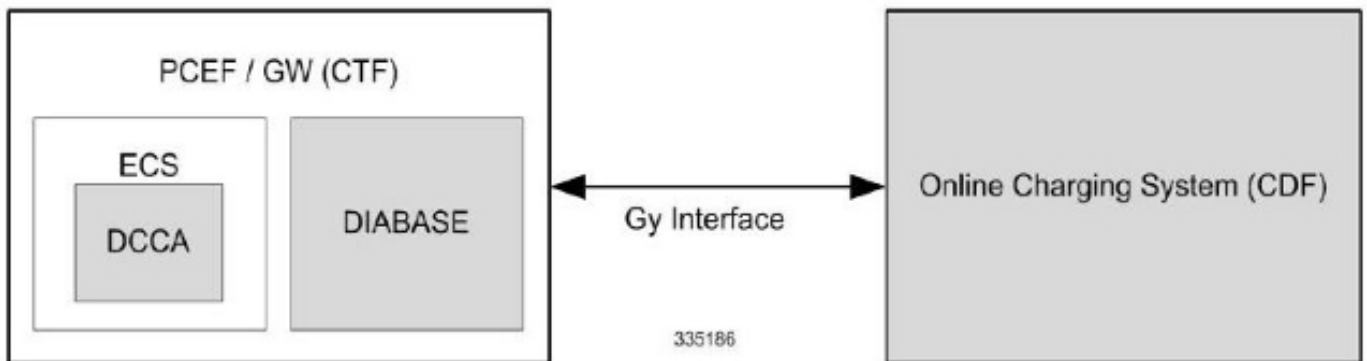
- SU メカニズム

設定

ここでは、FH メカニズムと SU メカニズムをサポートするために必要な設定について説明します。

ネットワーク図

このドキュメントの説明では、次のトポロジを使用します。



Tx-Expiry

これは、*diameter credit-control* 設定で設定可能な DCCA 用アプリケーションレベル タイマーです。値は 1 ~ 300 秒です。

以下が一例です。

```
[local]host_name(config-dcca)# diameter pending-timeout
```

応答タイムアウト

これは *diabase* タイムアウトであり、*Diameter Endpoint* 設定で設定できます。値は 1 ~ 300 秒です。

注：このタイマーに設定される値は、Tx-Expiry タイマーに使用する値よりも大きい値である必要があります。

以下が一例です。

```
[context_name]host_name(config-ctx-diameter)# response-timeout
```

Diameter セッション フェールオーバー

この機能は、Diameter クレジット制御セッション フェールオーバーを有効または無効にするために使用されます。このフェールオーバーにより、システムはプライマリ サーバに到達できなくなった場合にセカンダリ サーバを使用できます。これは *diameter credit-control* 設定で設定できます。

以下が一例です。

```
local]host_name(config-dcca)# diameter session failover
```

FH メカニズム

FH メカニズムは、Diameter セッション フェールオーバーが存在する場合にのみ動作します。FH によりシステムは、接続またはメッセージレベルのエラーが発生した場合に、セッションを続行してオフラインに切り替えるか、またはセッションを終了するかを選択できます。

注：FH は、デフォルトで有効であり設定されています。

FH メカニズムの設定

FH メカニズムの設定は *diameter credit-control* 設定で行うことができます。FH 設定で使用される構文を次に示します。

```
failure-handling { initial-request | terminate-request | update-request } { continue  
[ go-offline-after-tx-expiry | retry-after-tx-expiry ] | retry-and-terminate,  
[ retry-after-tx-expiry ] | terminate }
```

1 番目のセクションは要求タイプ (Initial (CCR-I)、Update (CCR-U)、および Terminate (CCR-T)) を指定します。

2 番目のセクションは、FH メカニズムがアクティブになったときに実行するアクションを指定します。FH メカニズムで指定できる 3 種類のアクションは次のとおりです。

- **continue** : セッションを続行でき、セッションがオフラインに切り替わります。この機能は、Tx-expiry に関連する 2 つのオプションを使用します。

go-offline-after-tx-expiry : Tx-expiry の発生後にオフライン課金を開始します。

retry-after-tx-expiry : Tx-expiry の発生後にセカンダリ サーバを再試行します。

- **retry-and-terminate** : システムによるセカンダリ サーバの再試行後、セカンダリ サーバも使用できない場合にセッションを終了します。このアクションでは **retry-after-tx-expiry** オプションも使用されます。このオプションでは Tx-expiry の発生後にセカンダリ サーバを再試行します。
- **terminate** : セカンダリ サーバとの通信を試行せずにセッションを終了します。

FH メカニズムのデフォルトの動作

ここでは、設定が行われていない場合の FH のデフォルトの動作について説明します。デフォルトでは、**terminate** アクションが設定されている場合を除き、FH メカニズムは応答タイムアウト (RT) 中にアクティブになります。

サーバから **Credit-Control-Failure-Handling** 属性値ペア (AVP) を受信すると、受信した設定が適用されます。

次に例を示します。

- **Initial-Request > Terminate**
- **Update-Request > Retry-and-Terminate**
- **Terminate-Request > Retry-and-Terminate**

FH メカニズムの詳細なコール フロー

ここでは、FH メカニズムの詳しいコール フローと有効なすべてのオプションについて説明します。

Initial-Request

CCFH の設定	CLI コマンド	Tx での動作	RT での動作	セカンダリが稼働している場合	セカンダリが稼働していない場合
	initial-request continue	N/A	[Continue]	セカンダリが RT の後に引き継ぐ	次の R... 。DCCA... (DCC...) セッショングループ... これ以...
[Continue]	initial-request continue go-offline-after-tx-expiry	オフライン	N/A	Tx でオフラインになる	Tx で...
	initial-request continue retry-after-tx-expiry	[Continue]	N/A	セカンダリが Tx の後に引き継ぐ	次の T...
Retry-and-terminate	initial-request retry-and-terminate	N/A	再試行	セカンダリが RT の後に引き継ぐ	次の R...

	initial-request				セカンダリ	
	retry-and-terminate	再試行	N/A		が	次の T
	retry-after-tx-expiry				Tx の後に引き継ぐ	
終了	initial-request	終了	N/A		Tx の後に終了する	Tx の後
	terminate					

Update-Request

CCFH の設定	CLI コマンド	Tx での動作	RT での動作	セカンダリが稼働している場合	セカンダリがダウンしている場合	
	update-request continue	N/A	[Continue]	セカンダリが RT の後に引き継ぐ		次の RT の後にオフラインになる
[Continue]	update-request continue go-offline-after-tx-expiry	オフライン	N/A	Tx でオフラインになる		Tx でオフラインになる
	update-request continue retry-after-tx-expiry	[Continue]	N/A	セカンダリが Tx の後に引き継ぐ		次の Tx の後にオフラインになる
Retry-and-terminate	update-request retry-and-terminate	N/A	再試行	セカンダリが RT の後に引き継ぐ		次の RT の後に CCR-T を送信する
	update-request retry-and-terminate retry-after-tx-expiry	再試行	N/A	セカンダリが Tx の後に引き継ぐ		次の Tx の後に CCR-T を送信する
終了	update-request terminate	終了	N/A	Tx の後に CCR-T を送信する		Tx の後に CCR-T を送信する

Terminate-Request

CCFH の設定	CLI コマンド	Tx での動作	RT での動作	セカンダリが稼働している場合	セカンダリがダウンしている場合	
	terminate-request continue	N/A	再試行	CCR-T がセカンダリに RT の後で送信される		次の RT の後に終了する
[Continue]	terminate-request continue go-offline-after-tx-expiry	再試行	N/A	CCR-T がセカンダリに Tx の後で送信される		次の Tx の後に終了する
	terminate-request continue retry-after-tx-expiry	再試行	N/A	CCR-T がセカンダリに Tx の後で送信される		次の Tx の後に終了する
Retry-and-	terminate-request	N/A	再試行	CCR-T が		次の RT の後に終了する

	retry-and-terminate			セカンダリに RT の後で送信される	
terminate	terminate-request retry-and-terminate retry-after-tx-expiry	再試行	N/A	CCR-T がセカンダリに Tx の後で送信される	次の Tx の後に終了する
終了	terminate-request terminate	終了	N/A	Tx の後に Tx の後に引き続く	Tx の後に終了する

SU メカニズム

SU メカニズムは FH メカニズムに似ていますが、よりきめの細かい障害コントロール手順が提供されます。メッセージレベルおよび接続レベル (トランスポート) の障害発生後にセッションを続行できるだけでなく、OCS からの応答に時間がかかる場合にもこのメカニズムを使用できます。一定期間またはクォータを使い切るまでセッションを続行してから終了するオプション、また、セッションがオフラインになるかまたは終了するまでの設定可能な暫定クォータ (ポリュームと時間) と設定可能なサーバ試行回数を使用するオプションがあります。

SU メカニズムの設定

SU メカニズムの設定は *diameter credit-control* 設定で行うことができます。SU の設定で使用する構文は、ご使用のバージョンに応じて異なります。

バージョン 12.1 以前で SU メカニズムの設定に使用される構文は次のとおりです。

```
servers-unreachable { initial-request { continue | terminate [ after-timer-expiry
<timeout_period> ] } | update-request { continue | terminate [ after-quota-expiry
| aftertimer-expiry <timeout_period> ] } }
```

バージョン 12.2 以降で SU メカニズムの設定に使用される構文は次のとおりです。

```
servers-unreachable { behavior-triggers { initial-request | update-request }
result-code { any-error | <result-code> [ to <end-result-code> ] }
| transport-failure [ response-timeout | tx-expiry ] | initial-request
{ continue [ { [ after-interim-time <timeout_period> ] [ after-interim-volume
<quota_value> ] } server-retries <retry_count> ] | terminate [ {
[ after-interim-time <timeout_period> ] [ after-interim-volume <quota_value> ]
} server-retries <retry_count> | after-timer-expiry <timeout_period> ] }
| update-request { continue [ { [ after-interim-time <timeout_period> ]
[ after-interim-volume <quota_value> ] } server-retries <retry_count> ]
| terminate [ { [ after-interim-time <timeout_period> ] [ after-interim-volume
<quota_value> ] } server-retries <retry_count> ] | after-quota-expiry |
after-timer-expiry <timeout_period> ] } }
```

注：バージョン 12.2 より前は、FH メカニズムと SU メカニズムを個別に設定できる柔軟性がありました。ただしバージョン 12.2 以降では、SU メカニズムと FH メカニズムが設定されている場合は SU メカニズムが FH メカニズムよりも優先されます。

サーバから CC-FH AVP が返され、一連の動作トリガーに対して SU メカニズムが設定されている場合、SU 設定が適用されます。それ以外の場合は、CC-FH AVP 値が適用されます。デフォルト

トでは、3002、3004、3005 などの結果コードは送信失敗に分類され、RTとして扱われます。

SU メカニズムで指定できるアクションは次のとおりです。

- **Behavior-Trigger** : メッセージのタイプ (Initial-Request (CCR-I) または Update-Request (CCR-U)) を指定します。このトリガーには次の 3 つのオプションを使用できません。

Result-Code : 特定の結果コード、結果コードの範囲 (3000 ~ 5999)、またはどのようなエラーとメッセージ タイプでも組み合わせて設定できます。

Transport-Failure この仕様は、バージョン12.0との下位互換性があるトランスポート障害時に動作をトリガーします。この設定には、次の2つのオプションがあります。

Response-Timeout : このオプションを指定すると、RT の発生時に動作がトリガーされます。このオプションは常にトランスポート障害で使用されます。

Tx-Expiry : このオプションを指定すると、Tx-expiry の発生時の動作がトリガーされます。このオプションは常にトランスポート障害で使用されます。

Actions : CCR-I または CCR-U の SU トリガーが発生した場合に実行するアクションを指定します。このアクションは、メッセージ タイプとソフトウェア バージョンによって異なります。

- **Continue** : セッションがオフラインに切り替わって続行します。バージョン12.2より前のバージョンでは、このアクションの使用に使用できるオプションはありません。バージョン 12.2以降では、暫定クォータ、サーバ再試行、タイマー期限切れ後のオプションを、このアクションで設定できます。
- **Terminate** : このアクションを指定すると、サーバが到達不能になった場合にセッションが終了します。このアクションでは、interim quota、server-retries、および after-timer-expiry オプションを指定できます。

次のオプションは、*Continue* アクションまたは *Terminate* アクションで使用できます。

- **after-interim-time** : このオプションでは、暫定タイムアウト期間の経過後にコールを続行または終了できます。これは、アクション実行前の時間クォータに似ています。値は秒単位であり、値の範囲は 1 ~ 4,294,967,295 です。
- **after-interim-volume** : このオプションでは、暫定クォータが割り当てられ、設定されたボリュームを使い切ったら、セッションを続行するかセッションを終了できます。値はバイト単位であり、値の範囲は 1 ~ 4,294,967,295 です。
- **server-retries** : このオプションでは、セッションを続行または終了する前に PCEF が OCS を再試行できます。再試行回数は 0 ~ 65,535の範囲で設定できます。値が0の場合、再試行は行われず、アクションは直ちに適用されます。

注 : *server-retries* オプションを使用する場合常には、*after-interim-time* オプションと *after-interim-volume* オプションも使用する必要があります。あるいはこの 3 つのオプションを同時に使用し、*Continue* アクションと *Terminate* アクションの両方に適用することができま

す。 *after-interim-time* オプションと *after-interim-volume* オプションでは、時間クォータとボリュームクォータが割り当てられ、どちらか最初に使い切ったクォータによってサーバ再試行がトリガーされます。

- **after-timer-expiry** : このオプションは、終了するまでセッションがオフラインステータスである期間 (秒単位) を指定します。値の範囲は1 ~ 4,294,967,295です。このオプションは、終了操作にのみ適用されます。
- **after-quota-expiry** : このオプションでは、すでに割り当てられているクォータを使い切った場合にセッションが終了します。

いくつかの重要な情報を次に示します。

- *after-interim-time*、*after-interim-volume*、および *server-retries* オプションは個別または組み合わせて使用でき、*Continue* アクションと *Terminate* アクションの両方に適用されます。
- *after-quota-expiry* オプションは、*Update-Requests* 動作トリガーのみに適用されます。
- *after-timer-expiry* オプションは *Terminate* アクションのみに適用されます。
- *after-interim-time*、*after-interim-volume*、および *server-retries* オプションは、バージョン 12.2 以降にのみ適用されます。
- *diameter* セッションフェールオーバーがサポートされ (かつ設定され) ている場合、常にセカンダリサーバとの通信が行われてから SU メカニズムがトリガーされます。
- 暫定時間クォータまたは暫定ボリュームクォータを使い切り、*server retries* オプションに 1 以上の値が設定されている場合、常に SU メカニズムがトリガーされる前に最後に通信したサーバと通信が行われます。たとえば OCS1 を最初に試行し、OSC1 でエラーが発生した後で OCS2 を試行する場合、OCS2 との通信により SU メカニズムがトリガーされます。サーバ再試行中に、最初に OCS2 との通信が行われ、OCS2 から否定的な応答があると OCS1 との通信が行われます。

SU メカニズムのコールフロー

SU メカニズムは、CCR-I または CCR-U の失敗によってトリガーされます。この原因としては、エラーコード、トランスポート障害、Tx-expiry、RT などがあります。アクションは、暫定クォータ (時間/ボリューム) の割り当て、サーバ再試行カウント、セッションがオフラインまたは終了するまでのタイマー値 (指定されている時間にわたってセッションがオフラインになる、終了のみ) またはクォータの期限切れ (CCR-U および終了のみ) のいずれかにできます。

暫定クォータは、Multiple Services Credit Control (MSCC) シナリオでのレーティンググループ (RG) 単位ではなく、セッション単位で指定されます。

プライマリサーバによってトランスポート障害が引き起こされ、セカンダリサーバが Tx-expiry または応答タイムアウトをトリガーする可能性があります。このシナリオでは、最後に発生したエラーが障害のトリガーとみなされます。

どの障害トリガーにも SU メカニズムが設定されていない場合、FH メカニズムがトリガーされます。

注：以降の項では、SU メカニズムに関連するコール フローの例を示します。これらのコール フローは、diameter-session-failover がサポートされており、セカンダリ サーバで Tx-expiry 値が RT 値未満に設定されていることを前提としています。また、トランスポート障害、Tx-expiry、および RT に対して SU メカニズムが設定されていることを前提としています。

セッションの接続解除が行われない Initial-Request

このシナリオのメッセージ フローを次に示します。

1. PCEF が CCR-I を OCS に送信します。
2. タイムアウトまたはトランスポート障害が検出されます。トランスポート障害が検出された場合、PCEF は即時にセカンダリ サーバで再試行します。それ以外の場合は Tx-expiry がトリガーされます。
3. セカンダリ サーバでもトランスポート障害またはタイムアウトが発生すると、SU メカニズムがトリガーされます。これは、トランスポート障害の場合は即時に発生し、タイムアウトの場合は Tx-expiry の後に発生します。
4. 暫定ボリュームまたは暫定時間が設定されている場合、暫定クォータがセッションに割り当てられます。
5. 暫定クォータ（時間またはボリューム）を使い切り、*server retries* にゼロよりも大きい値が設定されている場合、SU メカニズムをトリガーしたサーバに CCR-I が再度送信されます。さらに別の障害が発生すると、CCR-I が別のサーバに送信されます。
6. トランスポート障害または Tx タイムアウトが再び検出されたら、OCS から正常な応答が返されないことが *server retries* に指定された回数になるまで、ステップ 2 ~ 5 が繰り返されます。
7. それでも問題が解決しない場合、設定に基づいて、セッションが続行される（オンラインに切り替えられる）か、または終了します。

注：CCR-I が原因でセッションが SU モードになっている間に使いきった暫定クォータは、次の CCR-I では報告されません。使用された暫定クォータはすべて、正常に完了した CCA-I の後に続く CCR-U で報告されます。

セッションの接続解除が行われる Initial-Request

このシナリオのメッセージ フローを次に示します。

1. PCEF が CCR-I を OCS に送信します。
2. タイムアウトまたはトランスポート障害が検出されます。トランスポート障害が検出された場合、PCEF は即時にセカンダリ サーバで再試行します。それ以外の場合は Tx-expiry がトリガーされます。

3. セカンダリ サーバでもトランスポート障害またはタイムアウトが発生すると、SU メカニズムがトリガーされます。これは、トランスポート障害の場合は即時に発生し、タイムアウトの場合は Tx-expiry の後に発生します。
4. 暫定ボリュームまたは暫定時間が設定されている場合、暫定クォータがセッションに割り当てられます。
5. 暫定クォータ (時間またはボリューム) を使い切り、*server retries* にゼロよりも大きい値が設定されている場合、SU メカニズムをトリガーしたサーバに CCR-I が再度送信されます。さらに別の障害が発生すると、CCR-I が別のサーバに送信されます。
6. トランスポート障害または Tx タイムアウトが再び検出されたら、OCS から正常な応答が返されないことが *server retries* に指定された回数になるまで、ステップ 2 ~ 5 が繰り返されます。この時点で、暫定クォータすべてを使い切らずにセッションが接続解除されます。
7. セッション終了後には、新規セッション開始のため PCEF が CCR-I を再度送信します。これが成功すると、PCEF は CCR-T を送信し、使用された一時クォータ全体を報告します。

セッションの接続解除が行われない Update-Request

このシナリオのメッセージ フローを次に示します。

1. PCEF は OCS に CCR-U を送信します。
2. タイムアウトまたはトランスポート障害が検出されます。トランスポート障害が検出された場合、PCEF は即時にセカンダリ サーバで再試行します。それ以外の場合は Tx-expiry がトリガーされます。
3. セカンダリ サーバでもトランスポート障害またはタイムアウトが発生すると、SU メカニズムがトリガーされます。これは、トランスポート障害の場合は即時に発生し、タイムアウトの場合は Tx-expiry の後に発生します。
4. 暫定ボリュームまたは暫定時間が設定されている場合、暫定クォータがセッションに割り当てられます。
5. 暫定クォータ (時間またはボリューム) を使い切り、*server retries* にゼロよりも大きい値が設定されている場合、SU メカニズムをトリガーしたサーバに CCR-U が再度送信されます。さらに別の障害が発生すると、まだ報告されていない使い切ったクォータ全体が含まれている別のサーバに CCR-U が送信されます。
6. トランスポート障害または Tx タイムアウトが再び検出されたら、OCS から正常な応答が返されないことが *server retries* に指定された回数になるまで、ステップ 2 ~ 5 が繰り返されます。
7. 使用された暫定クォータ全体が、成功した CCR-U と共に OCS に報告されます。
8. それでも問題が解決しない場合、最大再試行回数まで再試行した後で、設定に基づいてセッションが続行される (オンラインに切り替えられる) か、または終了します。

セッションの接続解除が行われる Update-Request

このシナリオのメッセージ フローを次に示します。

1. PCEF は OCS に CCR-U を送信します。
2. タイムアウトまたはトランスポート障害が検出されます。トランスポート障害が検出された場合、PCEF は即時にセカンダリ サーバで再試行します。それ以外の場合は Tx-expiry がトリガーされます。
3. セカンダリ サーバでもトランスポート障害またはタイムアウトが発生すると、SU メカニズムがトリガーされます。これは、トランスポート障害の場合は即時に発生し、タイムアウトの場合は Tx-expiry の後に発生します。
4. 暫定ボリュームまたは暫定時間が設定されている場合、暫定クォータがセッションに割り当てられます。
5. 暫定クォータ (時間またはボリューム) を使い切り、*server retries* にゼロよりも大きい値が設定されている場合、SU メカニズムをトリガーしたサーバに CCR-U が再度送信されます。さらに別の障害が発生すると、まだ報告されていない使い切ったクォータ全体が含まれている別のサーバに CCR-U が送信されます。
6. トランスポート障害または Tx タイムアウトが再び検出されたら、OCS から正常な応答が返されないことが *server retries* に指定された回数になるまで、ステップ 2 ~ 5 が繰り返されます。この時点で、セッションは一時クォータをすべて使い果たす前に接続解除されます。
7. 使用済みクォータ全体を報告するため、PCEF は OCS に CCR-T を送信します。
8. OCS が結果コード 2002 で応答した場合は、追加レポートは不要です。

Update-Request と不明なセッション

このシナリオのメッセージ フローを次に示します。

1. PCEF は OCS に CCR-U を送信します。
2. タイムアウトまたはトランスポート障害が検出されます。トランスポート障害が検出された場合、PCEF は即時にセカンダリ サーバで再試行します。それ以外の場合は Tx-expiry がトリガーされます。
3. セカンダリ サーバでもトランスポート障害またはタイムアウトが発生すると、SU メカニズムがトリガーされます。これは、トランスポート障害の場合は即時に発生し、タイムアウトの場合は Tx-expiry の後に発生します。
4. 暫定ボリュームまたは暫定時間が設定されている場合、暫定クォータがセッションに割り当てられます。
5. 暫定クォータ (時間またはボリューム) を使い切り、*server retries* にゼロよりも大きい値が設定されている場合、SU メカニズムをトリガーしたサーバに CCR-U が再度送信されます。さらに別の障害が発生すると、まだ報告されていない使い切ったクォータ全体が含まれている別のサーバに CCR-U が送信されます。

6. OCS は、CCR-U に対して結果コード 5002 (不明なセッション ID) で応答します。これは、OCS が再起動し、セッション ID 情報が失われる場合に発生する可能性があります。

7. PCEF は CCR-I で新しいセッションを開始し、CCA-I を受信します。

8. PCEF は後続のメッセージで CCR-U を介して使用済み暫定クォータ全体を報告します。

Update-Request と複数の RG (MSCC シナリオ)

このシナリオのメッセージ フローを次に示します。

1. PCEF は RG1 の CCR-U を OCS に送信します。

2. タイムアウトまたはトランスポート障害が検出されます。トランスポート障害が検出された場合、PCEF は即時にセカンダリ サーバで再試行します。それ以外の場合は Tx-expiry がトリガーされます。

3. セカンダリ サーバでもトランスポート障害またはタイムアウトが発生すると、SU メカニズムがトリガーされます。これは、トランスポート障害の場合は即時に発生し、タイムアウトの場合は Tx-expiry の後に発生します。

4. 暫定ボリュームまたは暫定時間が設定されている場合、暫定クォータがセッションに割り当てられます。

5. この時点で RG2 も割り当てクォータをすべて使い果たしますが、セッションがすでに SU モードであり暫定クォータの使用を開始するため、CCR-U を開始しません。

6. 暫定クォータ (時間またはボリューム) を使い切り、*server retries* にゼロよりも大きい値が設定されている場合、SU メカニズムをトリガーしたサーバに CCR-U が再度送信されます。さらに別の障害が発生すると、両方の RG のまだ報告されていない使い切ったクォータ全体が含まれている別のサーバに、CCR-U が送信されます。

7. トランスポート障害または Tx タイムアウトが再び検出されたら、OCS から正常な応答が返されないことが *server retries* に指定された回数になるまで、ステップ 2 ~ 6 が繰り返されます。

8. 使用された暫定クォータ全体が、成功した CCR-U と共に OCS に報告されます。

9. それでも問題が解決しない場合、最大再試行回数まで再試行した後で、設定に基づいてセッションが続行される (オンラインに切り替えられる) か、または終了します。

Terminate-Request

このシナリオのメッセージ フローを次に示します。

1. PCEF は CCR-T を OCS に送信します。

2. タイムアウトまたはトランスポート障害が検出されます。トランスポート障害が検出された場合、PCEF は即時にセカンダリ サーバで再試行します。それ以外の場合は Tx-expiry がト

リガーされます。

3. セカンダリ サーバでもトランスポート障害またはタイムアウトが発生すると、セッションが削除されます。

CCR エラー コードの処理

このシナリオのメッセージ フローを次に示します。

1. PCEF は CCR を OCS に送信し、OCS がエラー コードで応答します。
2. エラー コードは SU メカニズムで静的に設定されています。
3. PCEF はセカンダリ サーバを再試行せずに暫定クォータを提供します。

FH および SU の設定例

ここでは、FH および SU メカニズムの設定例を示します。FH および SU メカニズムの両方が設定されている場合、同じ動作トリガーでは FH よりも SU が優先されます。

以下が一例です。

```
credit-control group test
diameter origin endpoint test
diameter peer-select peer test
quota volume-threshold percent 10
diameter pending-timeout 80 deciseconds msg-type any
diameter session failover
trigger type rat lac
apn-name-to-be-included virtual
quota request-trigger exclude-packet-causing-trigger
failure-handling initial-request continue retry-after-tx-expiry
servers-unreachable initial-request terminate after-interim-volume 200
after-interim-time 3600 server-retries 0
servers-unreachable behavior-triggers initial-request transport-failure
tx-expiry
servers-unreachable update-request continue after-interim-volume 200
after-interim-time 3600 server-retries 50
servers-unreachable behavior-triggers update-request transport-failure
tx-expiry
```

確認

設定が正しく機能するかどうかを確認するには、**show active-charging service <サービス名> コマンド**を入力します。

```
# show active-charging service name test
```

```
Service name: test
```

```
TCP Flow Idle Timeout : 300 (secs)
```

```
UDP Flow Idle Timeout : 300 (secs)
```

```
ICMP Flow Idle Timeout : 300 (secs)
```

```
ICMP Flow Idle Timeout : 300 (secs)
```

```
ALG Media Idle Timeout : 120 (secs)
```

```
TCP Flow-Mapping Idle Timeout : 300 (secs)
```

```
UDP Flow-Mapping Idle Timeout : Not Configured
```

```
Deep Packet Inspection: Enabled
```

```
Passive Mode : Disabled
```

```
CDR Flow Control : Enabled
```

```
CDR Flow Control Unsent Queue Size: 75
```

```
Unsent Queue high watermark: 56
```

```
Unsent Queue low watermark: 18
```

```
Content Filtering: Disabled
```

```
Dynamic Content Filtering: Disabled
```

```
URL-Blacklisting: Disabled
```

```
URL-Blacklisting Match-method: Exact
```

```
Content Filtering Match-method: Generic
```

Interpretation of Charging-rule-base-name: active-charging-group-of-ruledefs

Selection of Charging-rule-base AVP : Last

Credit Control:

Group : test

Mode : diameter

APN-name-to-be-included: gn

Trigger-Type : N/A

Failure-Handling:

Initial-Request : continue retry-after-tx-expiry

Update-Request : retry-and-terminate

Terminate-Request: retry-and-terminate

Server Unreachable Failure-Handling:

Initial-Request : terminate

Update-Request : continue

トラブルシューティング

SU および FH メカニズムに関連する統計情報を表示するには、**show active-charging credit-control statistics** コマンドを入力します。次に出力例を示します。

```
#show active-charging credit-control statistics
```

```
...
```

```
OCS Unreachable Stats:
```

```
Tx-Expiry: 2291985 Response-TimeOut: 615
```

```
Connection-Failure: 2 Action-Continue: 0
```

```
Action-Terminated: 0 Server Retries: 2023700
```

```
Assumed-Positive Sessions:
```


Current: 2 Cumulative: 2196851

この出力例に関する重要な注意点は次のとおりです。

- Tx-Expiry : Tx-expiry に起因する SU の状態を示します。
- Response-Timeout : RT に起因する SU の状態を示します。
- Connection-Failure : トランスポート障害に起因する SU の状態を示します。
- Action-Continue : このフィールドは、オフラインになったセッションの数を示します。
- Action-Terminate : このフィールドは、終了したセッションの数を示します。
- Server Retries : このフィールドは、OCS の再試行回数を示します。
- Assumed-Positive Sessions :

Current : このフィールドは、現在 SU 状態のセッションの数を示します。

Cumulative : このフィールドは、これまでに SU ステータスに移行したセッションの総数を示します。

セッションの SU 状態に関連する情報を表示するには、`show active-charging sessions full all` コマンドを入力します。次に出力例を示します。

```
#show active-charging sessions full all
```

```
..  
..
```

```
Current Server Unreachable State: CCR-I
```

```
Interim Volume in Bytes (used / allotted): 84/ 200
```

```
Interim Time in Seconds (used / allotted): 80/ 3600
```

```
Server Retries (attempted / configured): 1/ 50
```

この出力例に関する重要な注意点は次のとおりです。

- Current Server Unreachable State : 現在の SU 状態が CCR-I と CCR-U のいずれに起因するものであるかを示します。
- Interim Volume in Bytes (used/allotted) : 暫定ボリュームの使用バイト数と割り当てバイト数
を示します。
- Interim Time in Seconds (used/allotted) : 暫定時間の使用秒数と割り当て秒数
を示します。
- Server Retries (attempted/configured) : 実行されたサーバ再試行回数と設定されているサーバ再試行数を示します。

関連情報

- [『コマンドライン インターフェイス リファレンス、StarOS Release 16』](#)
- [テクニカル サポートとドキュメント - Cisco Systems](#)