



スタンドアロン Content Engine の透過リダイレクションの設定

この章では、コンテンツ要求をスタンドアロン Content Engine へ透過的にリダイレクトするための次の方法について説明します。

- Web Cache Communication Protocol (WCCP) 対応ルータでコンテンツ要求を代行受信し、スタンドアロン Content Engine にリダイレクトする
- レイヤ 4 スイッチでコンテンツ要求を代行受信し、スタンドアロン Content Engine にリダイレクトする

この章の内容は、次のとおりです。

- [WCCP 透過リダイレクションの概要 \(p.6-2\)](#)
- [WCCP 透過リダイレクションのスタンドアロン Content Engine の設定 \(p.6-8\)](#)
- [スタンドアロン Content Engine の WCCP フロー リダイレクションのディセーブル化および再イネーブル化 \(p.6-17\)](#)
- [スタンドアロン Content Engine における WCCP のシャットダウン \(p.6-18\)](#)
- [WCCP 透過リダイレクション用のルータの設定 \(p.6-19\)](#)
- [ルータでの WCCP サービスの設定 \(p.6-27\)](#)
- [ルータでの WCCP 統計情報の削除 \(p.6-37\)](#)
- [WCCP レイヤ 2 サポートの設定 \(p.6-37\)](#)
- [スタンドアロン Content Engine での WCCP サービスの設定例 \(p.6-42\)](#)
- [リダイレクション方式としてのレイヤ 4 スイッチングの設定 \(p.6-54\)](#)

Cisco ACNS 透過キャッシング ソリューションは、WCCP 対応ルータと各種の高度な技術を使用して、Web ブラウザが動作しない場合や、Web サーバが HTTP に準拠していない場合でも、Content Engine の透過性を保ちます。これらの技術の 1 つは、バイパス機能です。バイパス機能の設定方法については、「[スタンドアロン Content Engine でのバイパスの設定](#)」(p.15-3) を参照してください。



(注)

この章で使用する Content Engine CLI (コマンドライン インターフェイス) コマンドの構文および使用方法については、『*Cisco ACNS Software Command Reference*』 Release 5.4 を参照してください。

WCCP の詳細は、『*Cisco IOS Configuration Fundamentals Configuration Guide*』 および 『*Cisco IOS Configuration Fundamentals Command Reference*』 を参照してください。

WCCP 透過リダイレクションの概要

シスコは、ルータまたはスイッチが、IP、UDP、および TCP ヘッダー情報に基づいてパケットを代行受信し、これらのパケットを ACNS 5.x ソフトウェアを実行している Content Engine に透過的にリダイレクトできるように、Cisco IOS ソフトウェアに組み込む WCCP を開発しました。

WCCP には、Version 1 と Version 2 の 2 つのバージョンがあります。WCCP Version 1 は、1 つの WCCP サービス（標準 Web キャッシュ サービス [サービス 0] および単一ルータ）だけをサポートします。WCCP Version 1 の主な機能は、次のとおりです。

- 単一ルータだけをサポート（ホーム ルータ）
- トラフィック リダイレクションをポート 80 だけでサポート
- 各 WCCP サービスにつき、最大 32 の Content Engine をサポート
- バイパスはサポート対象外（スタティック バイパス、エラー バイパス、および認証バイパスはサポートされません）
- 応答の Generic Routing Encapsulation（GRE; 総称ルーティング カプセル化）は非サポート

WCCP Version 2 では、より多くの TCP ポートで、トラフィックを Content Engine にリダイレクトできます。以前は、TCP ポート 80 を宛先とする Web キャッシュ 情報だけをリダイレクトできましたが、多くのアプリケーションでは、他のポートを宛先とするパケットもリダイレクトする必要があります。たとえば、プロキシ Web キャッシュ 処理、FTP プロキシ キャッシング、ポート 80 以外に対する Web キャッシング、RealAudio、およびビデオなどのアプリケーションです。

複数のルータをはじめ、より多くの機能およびサービス（表 6-3 を参照）がサポートされるので、WCCP Version 2 の使用を推奨します。



(注) WCCP は IP ネットワークだけで動作します。

WCCP を使用したパケットのリダイレクション

WCCP 対応ルータによる透過リダイレクションを使用して要求を Content Engine にリダイレクトする場合、Web クライアントはコンテンツ要求をソースに送信しますが、要求が WCCP 対応ルータにより Content Engine にリダイレクトされることは認識しません。代行受信およびリダイレクトのプロセスは、コンテンツを要求しているクライアントからはまったく「見えない」または「透過的」で、デスクトップを変更する必要がないからです（クライアントは、ブラウザまたはメディア プレーヤーに特定のプロキシ サーバを指定する必要はありません）。Content Engine の動作はネットワークからは透過的なので、WCCP 対応ルータはリダイレクトされないトラフィックに対しては、通常の機能で動作します。

WCCP 透過リダイレクションを使用するには、最初に WCCP 対応ルータで WCCP サービスを定義する必要があります。特定サービスのためのパラメータは、サービスの名前、サービス ID（サービス番号）、および WCCP サービスのサポートに使用される ルータ インターフェイスです。

Content Engine WCCP の実装では、現在、すべての WCCP サービスに適用されるグローバル設定（ヒーリング パラメータ、スロー スタートなど）が可能です。マルチ サービス モデルでもその仕様は変更されず、該当する設定値は WCCP システム全体に対して、グローバルなまま維持されます。

Content Engine クラスタの一部である Content Engine のヒーリング モードの設定については、「Content Engine クラスタのヒーリング モードの設定」(p.7-76) を参照してください。WCCP スロー スタートなどの拡張キャッシング機能の設定については、第 15 章「スタンドアロン Content Engine の拡張透過キャッシング機能の設定」を参照してください。

Content Engine クラスタおよび WCCP 対応ルータを使用したパケットのリダイレクション

Content Engine のクラスタ（グループ）がある場合、すべての WCCP Version 2 対応ルータから認識され、IP アドレスが最小の Content Engine がリード Content Engine になります。リード Content Engine は、クラスタの Content Engine 間でどのようにトラフィックを割り当てるかを決定します。この割り当て情報は、指定されたリード Content Engine からサービス グループ全体に送られます。これにより、グループの WCCP 対応ルータはパケットを適切にリダイレクトでき、グループ内の Content Engine の負荷をより適切に管理できます。

次に、Content Engine クラスタ内の 1 台の Content Engine がどのようにリードに指定されるかを説明します。

1. 各 Content Engine が WCCP 対応ルータのリストを使用して設定されます。

WCCP Version 1 では、単一の WCCP 対応ルータが単一クラスタのサービスを提供し、そのルータがクラスタのデフォルト ホーム ルータになります。

WCCP Version 2 では、複数の WCCP 対応ルータで単一クラスタを処理できます（各ルータ リストに最大 8 台のルータを設定可能）。これにより、サービス グループ内で使用可能なルータにより、クラスタの各 Content Engine にパケットをリダイレクトできます。

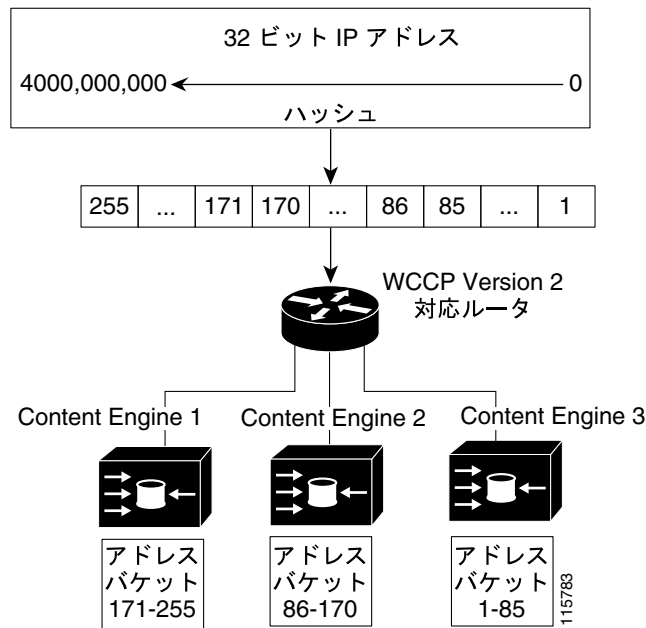
2. 各 Content Engine は、ルータ リスト上の各ルータに対して、存在を通知します。ルータは、サービス グループ内の Content Engine のビューを使用して応答します。
3. このビューがクラスタ内のすべての Content Engine 間で一致すると、1 台の Content Engine がリード Content Engine として指定され、WCCP 対応ルータがパケットのリダイレクトで配置する必要のあるポリシーを設定します。

WCCP Version 2 でのダイナミックな負荷分散の概要

WCCP Version 2 対応ルータは IP パケットを受信すると、そのパケットが、Content Engine にリダイレクトする必要がある要求かどうかを判別します。これは、要求と定義されたサービス基準とを照合することにより行われます。これらのパケットはルータの処理ルーチンに渡され、リダイレクトしたパケットを受信すべき Content Engine（ある場合）が判別されます。

代行受信されたパケットを受信する Content Engine は、宛先 IP アドレス上でハッシュ関数を実行し、パケットを割り当てるアドレス パケットを取得することにより、判別されます。これらのアドレス パケットは、Content Engine の数量と動作状況に応じて、特定の Content Engine にマップされます。図 6-1 を参照してください。

図 6-1 IP アドレスのハッシュによる負荷分散



(注) Content Engine で処理されないパケットは、パケットの送信先であるルータにトンネル伝送で戻されます。以前にリダイレクトしたパケットを受信した場合、ルータはそのパケットを再びリダイレクトしません。

WCCP Version 2 は動的な負荷分散をサポートしているので、ルータは、クラスタの各 Content Engine に転送する負荷を調整できます。WCCP は、2つの技術により、これを実現します。

- 負荷分散：この技術により、Content Engine に割り当てるハッシュ アドレス バケットを調整し、負荷容量が超過した Content Engine から、使用可能な容量のある他の Content Engine に負荷を移すことができます。
- 負荷制限：この技術により、WCCP 対応ルータは、Content Engine の容量を超過しないように負荷を選択的にリダイレクトします。

パケット転送方式について

WCCP 対応ルータは、2つのパケット転送方式を使用して、代行受信した要求をスタンドアロン Content Engine にリダイレクトします。

- GRE：Content Engine へのパス上のルータ数に関係なく、パケットを Content Engine に転送できます。
- レイヤ 2 リダイレクション：パケットをレイヤ 2 (MAC レイヤ) でスイッチングし、Content Engine に転送します。

表 6-1 に、これらの 2つの方式を示します。

表 6-1 パケット転送方式

パケット転送方式	負荷分散方式：ハッシュ	負荷分散方式：マスク
GRE (レイヤ 3)	パケットのリダイレクトは、ルータのソフトウェアによって完全に処理されます。	パケットリダイレクションは、ルータのソフトウェアによって処理されません。パケットの転送方式として GRE を使用する場合には、マスクの割り当ては推奨しません。
レイヤ 2 リダイレクション	最初にリダイレクトされるパケットはルータのソフトウェアによって処理され、以降にリダイレクトされるパケットはすべて、ルータのハードウェアによって処理されます。	すべてのパケットがルータのハードウェアによって処理されます (特殊ハードウェアが必要なため、現在サポートしているのは Catalyst 6000 シリーズのみ)。



(注)

どちらのパケット転送方式も、各種 WCCP サービス グループ内の Content Engine 間で、リダイレクトされたトラフィックがどのように負荷分散されるかは、ハッシュ パラメータによって指定されます。

アサイメント方式とは、Content Engine の間で負荷を分散するために、WCCP が使用する方式を意味します。負荷分散には、2つのアサイメント方式があります。ハッシュおよびマスクです。マスク負荷分散方式が指定されていない場合には、ハッシュ負荷分散方式 (図 6-1 を参照) がデフォルトの方式として使用されます。

このリダイレクション モードは、Content Engine によって制御されます。WCCP グループに参加する最初の Content Engine が、転送方式 (GRE またはレイヤ 2 リダイレクション) およびアサイメント方式 (ハッシュまたはマスク) を決定します。マスク アサイメントとは、WCCP レイヤ 2 Policy Feature Card 2 (PFC2) 入力リダイレクションを意味します。

転送方式のフォールバック機能に関しては、ルータが GRE リダイレクションを使用している場合でも (他の Content Engine が web キャッシュ サービスに GRE を使用している)、新しい Content Engine に (`wccp web-cache router-list-num 1 l2-redirect` グローバル コンフィギュレーション コマンドの指定により) レイヤ 2 リダイレクションを使用するように設定すると、Content Engine は レイヤ 2 リダイレクションのネゴシエートを試みます。Content Engine は レイヤ 2 リダイレクション機能をアドバタイズしますが、ルータが GRE で応答すると、Content Engine は GRE にフォールバックします。現時点では、Content Engine にレイヤ 2 リダイレクションの使用だけを設定することはできないので、これは予想される動作です。

アサイメント方式のフォールバック機能に関係なく、`assign-method-strict` オプション (`wccp https-cache assign-method-strict` コマンドを使用した `https-cache` サービスの `assign-method-strict` オプションの指定など) が使用されていない場合は、Content Engine は Content Engine クラスタ外に無制限に存続するのではなく、ハードウェアがサポートしているアサイメント方式にフォールバックします。マスクが WCCP 出力リダイレクションで選択される場合、Content Engine は、Multilayer Switch Feature Card (MSFC; マルチレイヤ スイッチ フィーチャ カード) および Policy Feature Card (PFC; ポリシー フィーチャ カード) で使用される元のハードウェア高速化にフォールバックします。

たとえば、WCCP Version 2 では、パケットをフィルタリングして、Content Engine から戻されたリダイレクト パケットと、戻されていないリダイレクト パケットを判別します。戻されたパケットはリダイレクトされません。これは、Content Engine が、そのパケットを処理しないことを判定済みであるからです。WCCP Version 2 は、Content Engine が処理しないパケットを、送信元であるルータに戻します。

Content Engine がパケットを拒否し、パケットを戻す主な理由は、次のとおりです。

- Content Engine が過負荷になり、パケットを処理するリソースがない。
- Content Engine が、サーバーエラーまたは認証の失敗により、自動バイパス機能をアクティブにした。この場合、クライアントはサーバに直接到達することができます。したがって、Content Engine はその障害の原因ではありません。
- Content Engine が、パケットの処理に逆効果を招く条件（たとえば、IP 認証がすでにオンになっている場合）でフィルタリングしている。
- Content Engine 上にスタティックバイパスリストを設定している。



(注)

パケットは、WCCP 対応ルータと Content Engine 間の接続の送信元にリダイレクトされます。そのアドレスは、使用される Cisco IOS ソフトウェアのバージョンに応じて、発信インターフェイスのアドレスか、ルータの IP アドレスのどちらかになります。元のアドレスがルータの IP アドレスである場合、そのルータ IP アドレスが Content Engine の WCCP 対応ルータ リストに保存されていることが重要です。ルータ リストの詳細は、「[スタンドアロン Content Engine のルータ リストの定義](#)」(p.6-12) を参照してください。

Cisco Express Forwarding (CEF) は、WCCP Version 2 に統合され、パケットのリダイレクト時に最適のパフォーマンスを実現します。さらに、WCCP Version 2 では複数のルータ (ルータ リスト) を設定して、特定の WCCP サービス (RTSP リダイレクションなど) をサポートできます。詳細は、「[WCCP 透過リダイレクションのスタンドアロン Content Engine の設定](#)」(p.6-8) を参照してください。サポート対象の WCCP Version 2 の機能とサービスについては、[表 6-3](#) を参照してください。

これらのパケット転送方式の詳細については、次を参照してください。

- [パケットの転送方式にレイヤ 3 GRE を使用する方法](#) (p.6-6)
- [パケットの転送方式にレイヤ 2 リダイレクションを使用する方法](#) (p.6-7)

パケットの転送方式にレイヤ 3 GRE を使用する方法

GRE は、レイヤ 3 の技術であり、データグラムを WCCP 対応ルータで IP パケットにカプセル化したあと、Content Engine へリダイレクトします (透過 プロキシサーバ)。この中間の宛先では、データグラムのカプセル化が解除され、キャッシュ ミスの場合に要求が満たされるように、オリジンサーバにルーティングされます。この過程で、オリジンサーバへのトリップは 1 ホップの内部データグラムとして扱われます。通常、GRE を使用して行われるリダイレクトトラフィックは、GRE トンネルトラフィックと呼ばれています。GRE では、すべてのリダイレクションはルータソフトウェアによって処理されます。

WCCP リダイレクションの場合、シスコ製ルータは TCP SYN パケットを宛先に転送しません。ルータには、接続の宛先ポートでイネーブルな WCCP があるためです。その代わりに、WCCP 対応ルータは、GRE トンネリングを使用してパケットをカプセル化し、WCCP 対応ルータからリダイレクトされるパケットを受信するように設定されたスタンドアロン Content Engine に送信します。

リダイレクトされたパケットを受信すると、Content Engine は次の動作を行います。

1. GRE レイヤをパケットから除去する。
2. このリダイレクトされたパケットを受け入れ、コンテンツの要求を処理する必要があるかを決定する。

- a. Content Engine がその要求を受け入れる場合、TCP SYN ACK パケットをクライアントに送信します。この応答パケットでは、Content Engine はソース アドレスとして指定されているオリジナルの宛先（オリジン サーバ）の IP アドレスを使用します。クライアントが Content Engine を認識しないように（透過的に）実行するために、Content Engine は、クライアントの TCP SYN パケットが到達する宛先のように認識されます。
- b. Content Engine は、その要求を受け入れないという決定をする場合、GRE の TCP SYN パケットを再度カプセル化し WCCP 対応ルータに返信します。ルータは、この場合、Content Engine が接続しないことを認識し、そのパケットをオリジナルの宛先（つまり、オリジン サーバ）に転送します。

たとえば、一連の特定のクライアントからの、または一連の特定のサーバ宛での要求をバイパスするように設定されている場合、スタンドアロン Content Engine は、その要求を受け入れないと判断します。

パケットの転送方式にレイヤ 2 リダイレクションを使用する方法

レイヤ 2 リダイレクションとは、ルータまたはスイッチ上の WCCP が、レイヤ 2 のルータ ハードウェアで WCCP のトラフィック代行受信、およびリダイレクション機能を部分的または全部を実装しているスイッチング ハードウェアを利用する場合の用語です。現在、このリダイレクションタイプをサポートしているのは、Cisco Catalyst 6000 シリーズ、および 6500 シリーズのスイッチ製品だけです。レイヤ 2 リダイレクションでは、最初にリダイレクトされるトラフィック パケットは、ルータ ソフトウェアにより処理されます。それ以降のトラフィックは、ルータ ハードウェアで処理されます。レイヤ 2 リダイレクションでは、Content Engine はルータまたはスイッチに対して、特定パケット フィールドにビットマスクを適用するように指示します。これにより、クラスタ内の Content Engine に、マスクの結果またはインデックスが、マスク インデックス アドレス テーブルの形式でマップされます。このリダイレクション プロセスはスイッチング ハードウェアで加速されるので、レイヤ 2 リダイレクションはレイヤ 3 GRE よりも効率的です。



(注)

WCCP は Content Engine に対してのみライセンスされており、リダイレクトするルータにはライセンスされていません。WCCP は、通常のルータまたはスイッチの動作に干渉することはありません。

レイヤ 2 リダイレクションの設定については、「[WCCP レイヤ 2 サポートの設定](#)」(p.6-37) を参照してください。

WCCP 透過リダイレクションのスタンドアロン Content Engine の設定

スタンドアロン Content Engine および WCCP 対応ルータのサポート対象の WCCP サービスのタイプは、表 B-3 に示すように、WCCP Version 1 または Version 2 のどちらを使用するかによって異なります。標準 Web キャッシュ サービス（サービス 0）を除くすべてのサービスでは、特定の WCCP サービスをサポートするために、ルータおよびスタンドアロン Content Engine 上で（WCCP Version 1 ではなく）WCCP Version 2 を実行する必要があります。表 6-2 に、スタンドアロン Content Engine でサポートされる WCCP 透過リダイレクション サービスを示します。

表 6-2 サポート対象の WCCP 透過リダイレクション サービス

WCCP 透過リダイレクション サービス	WCCP サービス名	説明	詳細
サービス 80	rtsp	RealMedia クライアントから Content Engine に RTSP 要求をリダイレクトするために使用される RTSP リダイレクション サービス。RealMedia 要求を透過的にリダイレクトするには、サービス 80 だけを設定します。WMT RTSP 透過リダイレクション（Windows Media 9 プレーヤーからの RTSP 要求の透過リダイレクション）をサポートするには、サービス 80 および サービス 83 の両方を設定する必要があります。	クライアントが RealMedia プレーヤーの場合、バックエンド RTSP サーバは RealProxy サーバです。クライアントが Windows Media 9 プレーヤーの場合、バックエンド RTSP サーバは、Content Engine 上で稼働している Windows Media 9 サーバです。
サービス 81	mmstu	WMT クライアントから Content Engine に MMS 要求（MMS 要求および MMS-over-HTTP 要求）をリダイレクトするために使用される MMSTU リダイレクション サービス。	クライアントは WMT クライアント（Windows Media プレーヤー Version 6、7、または 9 など）で、バックエンド MMS サーバは WMT サーバ（mms_server プロセス）です。
サービス 82	mmsu	WMT クライアントから Content Engine に MMS 要求（MMS 要求および MMS-over-HTTP 要求）をリダイレクトするために使用される MMSU リダイレクション サービス。WMT 要求の MMS リダイレクションを実行するには、ルータ上にサービス 81 および 82 を設定する必要があります。	クライアントは WMT クライアント（Windows Media プレーヤー Version 6、7、または 9 など）で、バックエンド MMS サーバは WMT サーバ（mms_server プロセス）です。
サービス 83	wmt-rtspu	WMT クライアント（Windows Media 9 プレーヤー）から（Windows Media 9 サーバとし機能している）Content Engine への RTSP 要求をリダイレクトするために使用される WMT RTSP リダイレクション サービス。	クライアントおよびサーバは WMS 9 です（Windows Media 9 プレーヤー、および Content Engine 上で稼働し、WMT RTPS 要求のバックエンド RTSP サーバとして機能している Windows Media Series 9 サーバなど）。

スタンドアロン Content Engine で設定可能な WCCP オプションおよびサービスのリストを表示するには、`wccp EXEC` コマンドに、疑問符（「?」）を付けて入力します。次に、WCCP Version 2 がイネーブである Content Engine の出力例を示します。WCCP Version 1 を使用している場合、1 台のルータ（ホーム ルータ）と 1 つの WCCP サービス（標準の Web キャッシュ サービス）だけがサポートされています。


```

ContentEngine(config)# wccp ?
access-list          Configure an IP access-list for inbound WCCP encapsulated traffic
custom-web-cache    Custom web caching service
dns                  Caching Domain Name Service
flow-redirect        Redirect moved flows
ftp-native           Transparent FTP proxy caching service
home-router          WCCP Version 1 Home Router Ip address
https-cache          HTTPS caching service
port-list            Port list for use in WCCP service
reverse-proxy        Reverse Proxy web caching service
router-list          Router List for use in WCCP services
rtsp                 RTSP protocol transparent interception
service-number       WCCPv2 service number
shutdown             Wccp Shutdown parameters
slow-start           accept load in slow-start mode
spooof-client-ip     Use client IP while connecting to the origin server
version              WCCP Version Number
web-cache            Standard web caching service
wmt                  Windows media caching service
wmt-rtspu            Windows media RTSPU (port 5005) transparent interception

```

WCCP Version 1 を使用した標準 Web キャッシュ サービス (サービス 0) の設定の詳細は、「例 1 — WCCP Version 1 を使用した Web キャッシュ サービスの設定」(p.6-43) を参照してください。

WCCP Version 2 のオプションとサービスについては、表 6-3 を参照してください。表 6-3 のアスタリスク (「*」) は、各機能が WCCP オプションまたは WCCP サービスのどちらかであることを示しています。

表 6-3 WCCP Version 2 のオプションとサービス

WCCP のオプション またはサービス名	WCCP オプション	WCCP サービス	詳細
access-list	*		スタンドアロン Content Engine での WCCP アクセス リストの設定
custom-web-cache		*	ルータでのカスタム Web キャッシュ サービス (サービス 98) の設定
dns		*	ルータでの DNS キャッシング サービス (サービス 53) の設定
flow-redirect	*		スタンドアロン Content Engine の WCCP フローリダイレクションの ディセーブル化および再イネーブル化
ftp-native		*	ルータでの FTP-Native キャッシング サービス (サービス 60) の設定
https-cache		*	ルータでの HTTPS キャッシュ サービス (サービス 70) の設定
port-list	*		スタンドアロン Content Engine のポートリストの定義
reverse-proxy		*	ルータでのリバース プロキシ サービス (サービス 99) の設定
rtsp		*	ルータでの RTSP サービス (サービス 80) の設定
service-number	*		ルータでのユーザ定義 WCCP サービス (サービス 90 ~ 97) の設定
shutdown	*		スタンドアロン Content Engine のポートリストの定義
slow-start	*		WCCP スロー スタートの設定
spooof-client-ip	*		WCCP IP スプーフィングの設定
version	*		スタンドアロン Content Engine の WCCP のイネーブル化
web-cache		*	例 2 — WCCP Version 2 を使用した Web キャッシュ サービスの設定
wmt		*	ルータでの MMST サービス (サービス 81) の設定 ルータでの MMSU サービス (サービス 82) の設定
wmt-rtspu		*	ルータでの WMT-RTSPU サービス (サービス 83) の設定



(注)

ACNS ソフトウェア リリース 5.2 以前の場合、WCCP Version 2 対応ルータおよび Content Engine でサポートされるアクティブ WCCP サービスは、最大 8 個です。ACNS ソフトウェア リリース 5.2.1 以降では、最大 25 のアクティブ WCCP Verison 2 サービスがサポートされます。ACNS ソフトウェア リリース 5.3.1 には、wmt-rtspu (サービス 83) が追加されています。ACNS ソフトウェア リリース 5.3.1 では、現在、18 のサービスを設定できます。

ACNS ソフトウェア 5.2 には、https-cache サービスに、別の代行受信モード (accept-all モード) が追加されています。このモードは、HTTPS トラフィックのフィルタリングをサポートするために追加されました。このモードは、従来の WCCP サービスと同様に機能します (デフォルトですべての Web トラフィックを代行受信する標準 Web キャッシュ サービス [サービス 0] など)。**wccp https-cache accept-all** グローバル コンフィギュレーション コマンドを使用すると、HTTPS キャッシュ (https-cache キャッシュ サービスが設定され、イネーブルである Content Engine) は、accept all モード (すべての HTTPS トラフィックが Content Engine により代行受信される) で動作します。それ以外の場合、HTTPS キャッシュは ACNS 5.1.x ソフトウェアと同様に、accept only モードで動作します。

透過リダイレクション用に WCCP を使用するには、Content Engine を適切に設定する必要があります。次の重要事項に注意してください。

- Content Engine は、1 台または複数の WCCP 対応ルータからリダイレクトされたパケットを受け入れるように設定する必要があります。WCCP Version 1 の場合、単一ルータ (ホーム ルータ) だけがサポートされます。WCCP Version 2 の場合、複数のルータ (ルータ リスト) がサポートされます。
- Content Engine 上のソフトウェアのバージョンは、WCCP 対応ルータ上にインストールされているバージョンとの互換性が必要です。
- Content Engine のパケットは暗号化または圧縮されず、内部の Network Address Translation (NAT; ネットワーク アドレス変換) ファイアウォール (存在する場合) の一部である必要があります。
- Web キャッシュ リダイレクト対応インターフェイスを越えて、サーバへのルート上に Content Engine を配置すると、IP ルート キャッシュにエントリが読み込まれません。
- ルータ上で WCCP を使用可能にした後、そのルータおよび Content Engine を透過キャッシング サービス用に設定する必要があります。

WCCP 透過リダイレクション用にスタンドアロン Content Engine を設定する方法については、次を参照してください。

- [スタンドアロン Content Engine の WCCP のイネーブル化 \(p.6-11\)](#)
- [スタンドアロン Content Engine の ポート リスト の定義 \(p.6-11\)](#)
- [スタンドアロン Content Engine の ルータ リスト の定義 \(p.6-12\)](#)
- [スタンドアロン Content Engine の WCCP サービスの設定 \(p.6-13\)](#)
- [スタンドアロン Content Engine の WCCP 設定情報の表示 \(p.6-16\)](#)

スタンドアロン Content Engine で WCCP サービスを設定する方法のサンプル シナリオについては、「[スタンドアロン Content Engine での WCCP サービスの設定例](#)」(p.6-42) を参照してください。ルータで WCCP 透過代行受信を設定する方法については、「[WCCP 透過リダイレクション用のルータの設定](#)」(p.6-19) を参照してください。

スタンドアロン Content Engine の WCCP のイネーブル化

スタンドアロン Content Engine で WCCP をイネーブルにするには、**wccp version** グローバル コンフィギュレーション コマンドを入力します。スタンドアロン Content Engine で実行する WCCP のバージョンを指定します。WCCP 環境で使用するルータのソフトウェアバージョンが、スタンドアロン Content Engine に設定された WCCP バージョンをサポートしていることを確認します。

次に、スタンドアロン Content Engine で WCCP Version 2 をイネーブルにする例を示します。

```
Content Engine (config)# wccp version 2
```

次に、スタンドアロン Content Engine で WCCP Version 1 をイネーブルにする例を示します。

```
Content Engine (config)# wccp version 1
```

Content Engine で一度にイネーブルにできるのは、WCCP の 1 つのバージョンだけです。より広範囲の WCCP オプションおよびサービスをはじめ、複数のルータ（ルータ リスト）がサポートされるので、WCCP Version 2 を実行することを推奨します。WCCP Version 2 のサポート対象の機能およびサービスについては、[表 6-3](#) を参照してください。

WCCP Version 2 をイネーブルにするときは、事前に WCCP Version 1 をディセーブルにする必要はありません。その逆も同様です。ただし、Content Engine 上で WCCP を適切にシャットダウンするには、現在実行中のバージョンをディセーブルにする必要があります。これについては、「[スタンドアロン Content Engine の WCCP フロー リダイレクションのディセーブル化および再イネーブル化](#)」(p.6-17) を参照してください。

スタンドアロン Content Engine の ポート リスト の定義

WCCP Version 1 の場合、TCP ポート 80 を宛先とする Web キャッシュ情報だけを Content Engine にリダイレクトできますが、多くのアプリケーションでは、他のポートを宛先とするパケットもリダイレクトする必要があります。たとえば、プロキシ Web キャッシュ処理、ポート 80 以外のポートに対する Web キャッシング、RealAudio、およびビデオなどのアプリケーションです。ルータが WCCP Version 1 ではなく WCCP Version 2 用に設定されている場合には、WCCP 対応ルータにポート 80 以外の TCP ポートを設定して、トラフィックを Content Engine にリダイレクトできます。

最大 8 つのポート リストを設定できます（ポート リスト番号 1～8）。これらのポート リストで、着信 WCCP リダイレクトトラフィックを待ち受ける、Content Engine 上のポート番号を指定します。ポート リストにより、Content Engine が着信 WCCP 要求を複数のポートで待ち受けるように設定できます。

デフォルトで、Content Engine はポート 80 で着信トラフィックを待ち受けます。作成する 8 つのユーザ定義 WCCP サービス（サービス 90～97）ごとに、1 つのポート リストを作成します。最大 8 つのポートを、ポート リスト単位で定義できます。この場合、各ポート リストに 1 つのポートが設定されます（ポート リスト 1 にポート 32 を設定するなど）。

```
ContentEngine(config)# wccp port-list 1 32
ContentEngine(config)# wccp port-list 2 33
ContentEngine(config)# wccp port-list 3 34
ContentEngine(config)# wccp port-list 4 35
ContentEngine(config)# wccp port-list 5 36
ContentEngine(config)# wccp port-list 6 37
ContentEngine(config)# wccp port-list 7 38
ContentEngine(config)# wccp port-list 8 39
```

スタンドアロン Content Engine の ルータ リスト の定義

Content Engine での WCCP Version 2 サービスの設定の一環として、Content Engine に対して特定の WCCP Version 2 サービス (rtsp サービスなど) をサポートする WCCP Version 2 対応ルータのリストを作成する必要があります。

スタンドアロン Content Engine でルータ リストを作成するには、**wccp router-list** グローバル コンフィギュレーション コマンドを使用します。Content Engine の特定の WCCP サービスをサポートする各 WCCP Version 2 対応ルータの IP アドレスを入力します。各種 WCCP サービスに対して複数の異なるルータを使用する場合、複数のルータ リストを作成する必要があります。各ルータ リストに、最大 8 のルータを含めることができます。

次に、ルータ リスト番号 1 を作成し、単一ルータ (IP アドレス 10.10.10.1 の WCCP Version 2 対応ルータ) を設定する例を示します。

```
ContentEngine(config)# wccp router-list 1 10.10.10.1
```

次に、ルータ リスト (ルータ リスト 1) を作成し、Content Engine がルータ リスト 1 上で WCCP Version 2 対応ルータからリダイレクトされる WMT トラフィック (「wmt」という名前の WCCP サービス) を受け入れるように設定する例を示します。

```
ContentEngine(config)# wccp router-list 1 10.10.10.2
ContentEngine(config)# wccp wmt router-list 1
ContentEngine(config)# wccp version 2
```

WCCP Version 1 では、標準 Web キャッシュ サービス (サービス 0) をサポートする単一の WCCP 対応ルータだけを設定できます。Content Engine のクラスタがある場合でも、単一の WCCP Version 1 対応ルータだけが Content Engine のクラスタと通信し、クラスタのデフォルト ホーム ルータになります。WCCP Version 1 を使用する標準 Web キャッシュ サービスのホーム ルータとしてのルータの設定方法については、「例 1 — WCCP Version 1 を使用した Web キャッシュ サービスの設定」(p.6-43) を参照してください。



(注)

ルータ リストに含まれる各ルータで WCCP をイネーブルにするためには、**ip wccp** グローバル コンフィギュレーション コマンドを使用する必要があります。

WCCP Version 2 用に Content Engine を設定する場合、Content Engine 上に、ルータ リストではなく IP マルチキャスト アドレスを設定できます。Content Engine 上でルータ リストを使用する場合には、IP マルチキャストを使用する必要はありませんが、各 Content Engine での設定作業がより多くなります。IP マルチキャスト アドレスを使用すると、Content Engine の設定作業およびプロトコル オーバーヘッドを削減できます。

IP マルチキャストでは、Content Engine に IP マルチキャスト アドレスを設定します。WCCP Version 2 対応ルータは、1 つまたは複数のインターフェイスで IP マルチキャスト アドレスを受信するように設定されます。これらのルータは、リダイレクトされた要求を Content Engine 上の指定された IP マルチキャスト アドレスに送信します。マルチキャスト アドレスは 224.0.0.0 ~ 239.255.255.255 でなければなりません。IP マルチキャスト アドレスの割り当ては、Internet Assigned Numbers Authority (IANA) によって管理されています。IANA は、IP マルチキャスト用に IPv4 クラス D アドレス スペースを割り当てています。このため、すべての IP マルチキャスト グループ アドレスは、224.0.0.0 ~ 239.255.255.255 の範囲内にあります。ただし、一部の送信元とグループ アドレスの組み合わせは、マルチキャストには使用できません。使用できないマルチキャスト アドレス範囲、および使用できない理由については、「使用できないマルチキャスト アドレスの割り当て」(p.B-16) を参照してください。

IP マルチキャストを次のように使用する場合には、サービス グループのメンバーとなる WCCP Version 2 対応ルータに、追加設定が必要です。

- サービス グループで使用する IP マルチキャスト アドレスを設定する必要がある。
- WCCP Version 2 対応ルータが IP マルチキャスト アドレスを受信するインターフェイスを、**ip wccp {web-cache | service-number} group-listen** コマンドで設定する必要がある。

ターゲット ルータに到達するために別のルータを通過する必要があるネットワーク構成の場合、通過するルータを設定して、次のような IP マルチキャスト ルーティングを行う必要があります。

- **ip multicast-routing** コマンドを使用し、ルータの IP マルチキャスト ルーティングを設定することにより、それをイネーブルにする必要がある。
- **ip pim** コマンドを使用して、Content Engine の接続先のルータ インターフェイスでマルチキャストを受信するように設定する必要がある。

スタンドアロン Content Engine の WCCP サービスの設定

WCCP 対応ルータと Content Engine のサポート対象 WCCP サービスには、一般的な基準セットおよび事前定義されたサービス ID (標準 web キャッシュ サービス [サービス 0] など) が含まれています。これらのサービスを、*事前定義 WCCP サービス* と呼びます。他の例には、reverse-proxy キャッシング サービス (サービス 99) https-caching サービス (サービス 70)、および rtsp サービス (サービス 80) があります。

一般的ではない他の WCCP サービスは、基準セットを指定し、ユーザ定義 WCCP Version 2 サービスにサービス ID を割り当てることによって定義します。WCCP Version 2 では、最大 8 つのユーザ定義 WCCP サービス (サービス 90 ~ 97) を定義できます。各ユーザ定義サービスは、最大 8 つのポートをサポートします。

ユーザ定義 WCCP サービスをサポートするスタンドアロン Content Engine の設定方法の詳細については、「[ユーザ定義 WCCP サービスをサポートするためのスタンドアロン Content Engine の設定](#)」(p.6-14) を参照してください。



(注) サポート対象の WCCP サービスについては、[表 B-3](#) を参照してください。

スタンドアロン Content Engine の WCCP サービスを設定するには、次の重要事項に注意してください。

- WCCP Version 1 でサポートされるのは、単一の WCCP サービス (標準 web キャッシュのサービス [サービス 0])、および単一のルータだけです。したがって、より広範囲の機能とサービス、および複数のルータがサポートされる WCCP Version 2 の使用を推奨します。
- スタンドアロン Content Engine で使用する特定の WCCP サービスをサポートする WCCP Version 2 対応ルータのリストを作成する必要があります。各種 WCCP サービスに対して複数の異なるルータを使用する場合、複数のルータ リストを作成する必要があります。
- また、ルータ リスト内の各ルータ上で、WCCP Version 2 をイネーブルにする必要があります。さらに、特定の WCCP サービスに対応付けた特定のルータ リスト内の各ルータ上で、指定した WCCP サービスをイネーブルにする必要があります。
- WCCP Version 2 をスタンドアロン Content Engine 上でイネーブルにし、このサービスをサポートする WCCP Version 2 対応ルータのリストを定義したあと、Content Engine および特定の WCCP サービスをサポートする WCCP 対応ルータ上で、指定した WCCP サービスをイネーブルにする必要があります。

次に、WCCP Version 2 対応ルータ上で、ユーザ定義 WCCP サービス（サービス 91）をイネーブルにする例を示します。

```
Router# configure terminal
Router(config)# ip wccp 91
Router(config)# interface ethernet 0
Router(config-if)# ip wccp 91 redirect out
```

- WCCP 変数に関連付けられた値を表示するには、WCCP Version 2 対応ルータ上で **show ip wccp EXEC** コマンドを使用します。
- クラスタと通信するルータの Time To Live (TTL) 値は、15 秒以下でなければなりません。
- サービス グループには、最高 32 台の Content Engine と 32 台の WCCP 対応ルータを組み込むことができます。
- クラスタ内のすべての Content Engine に、そのクラスタと通信するすべての WCCP 対応ルータが設定されている必要があります。クラスタ内のある Content Engine にルータがまったく設定されていない場合、サービス グループは不整合を検出し、その Content Engine は、サービス グループ内で動作しません。



(注)

Content Engine と WCCP 対応ルータをファイアウォールで隔てることはできません。ファイアウォールでは、オリジン Web サーバ宛ての packets トラフィックだけを処理し、そのサーバに代わって Content Engine がクライアントに送信する packets トラフィックは処理しないからです。

また、ほとんどの WCCP Version 2 機能では、**wccp** グローバル コンフィギュレーション コマンドで特定のオプションを設定する必要があります。**wccp** グローバル コンフィギュレーション コマンドの詳細は、『Cisco ACNS Software Command Reference』Release 5.4 を参照してください。ルータまたはスイッチの設定方法については、各デバイスに付属のソフトウェア マニュアルを参照してください。WCCP Version 2 コマンドおよびルータ設定例の詳細は、Cisco IOS ソフトウェアのオンライン マニュアルを参照してください。

ユーザ定義 WCCP サービスをサポートするためのスタンドアロン Content Engine の設定

ユーザ定義 WCCP サービスは、トラフィックをスタンドアロン Content Engine にリダイレクトするようにポート番号を設定できる WCCP Version 2 のサービスです。Content Engine は、透過的なフォワードプロキシサーバとして機能しています。

WCCP Version 2 では、複数のポートをサポート (WCCP サービスごとに最大 8 つのポート) する最大 8 つのユーザ定義 WCCP サービス (サービス 90 ~ 97) を設定できます。これらのサービスを設定するため、使用する各サービスに対して 1 つのポート リストを作成する必要があります (たとえば、サービス 90 に対してはポート リスト番号 1 を作成します)。ポート リストには、特定のユーザ定義 WCCP サービスで WCCP Version 2 対応ルータが WCCP リダイレクションをサポートするポート番号を指定します。ユーザ定義 WCCP サービスを設定する場合、Content Engine 上のキャッシング アプリケーション、HTTP キャッシング アプリケーション、またはストリーミング アプリケーションのどれにトラフィックをリダイレクトするかを指定する必要があります。

Content Engine で複数のポートを使用して Web トラフィックをキャッシングするには、Content Engine および WCCP Version 2 対応ルータが、ユーザ定義 WCCP サービスを実行するように設定します。通常、単一ポートだけをサポートする標準 WCCP サービス (事前定義の **https-cache**、**rtsp**、**mmst**、および **reverse-proxy** のサービスなど) について、HTTP、MMS、HTTPS、および RTSP 要求の WCCP リダイレクションを複数のポート上 (各サービスに最大 8 ポート) でサポートするには、これらのユーザ定義 Web サービスを使用します。

ユーザ定義 WCCP サービスは、Content Engine GUI または CLI コマンドを使用して、Content Engine 上で設定できます。Content Engine GUI では、**WCCP > Services** を選択します。表示された Services ウィンドウを使用して Content Engine を設定し、ユーザ定義の WCCP サービス (サービス 90 ~ 97) をサポートします。Services ウィンドウの使用の詳細については、ウィンドウの **HELP** ボタンをクリックしてください。

Content Engine CLI では、**wccp service-number** グローバル コンフィギュレーション コマンドを使用して Content Engine を設定し、ユーザ定義 WCCP サービスをサポートします。次の例のように、作成する各ユーザ定義の WCCP サービス (サービス 95 など) のマスクとルータ リストを定義できます。

```
ContentEngine(config)# wccp service-number 95 ?
mask                Specify mask used for CE assignment
router-list-num     Router list number
```

Content Engine が指定の WCCP サービスによってリダイレクトされた要求を受け入れる WCCP Version 2 対応ルータのリストを指定するには、**router-list-num** オプションを使用します。

```
ContentEngine(config)# wccp service-number 95 router-list-num 1
```

Content Engine が、指定の WCCP サービスによってリダイレクトされたトラフィックを待ち受けるポートを指定するには、**port lists** オプションを使用します。次に、Content Engine が、ルータ リスト 1 の WCCP Version 2 対応ルータからサービス 95 によりリダイレクトされたトラフィックを、ポート リスト番号 5 にリストされたポート上で待ち受けるように設定する例を示します。

```
ContentEngine(config)# wccp service-number 95 router-list-num 1 port-list-num 5
```

リダイレクトされたトラフィックの宛先を、Content Engine 上のキャッシング アプリケーション、HTTP キャッシング アプリケーション、またはストリーミング アプリケーションのどれにするかを指定するには、**applications** オプションを使用します。

```
ContentEngine(config)# wccp service-number 95 router-list-num 1 port-list-num 1
application ?
cache          Direct traffic to the caching application
https-cache   Direct traffic to the https caching application
streaming     Direct traffic to the streaming media application
```

Content Engine GUI を使用して、スタンドアロン Content Engine の WCCP のイネーブル化および設定を行う場合には、次の Content Engine GUI ウィンドウの各サービスで、指定されたルータ リストを指定する必要があります。Web Cache ウィンドウ (**WCCP > Web Cache**)、Reverse Proxy ウィンドウ (**WCCP > Reverse Proxy**)、Custom Web Cache ウィンドウ (**WCCP > Custom Web Cache**)、および WCCP Services ウィンドウ (**WCCP > Services**)。

スタンドアロン Content Engine でユーザ定義 WCCP サービス (サービス 90 ~ 97) を設定する方法の例については、「例 4 — スタンドアロン Content Engine での複数の WCCP Version 2 サービスの設定」(p.6-49) を参照してください。

スタンドアロン Content Engine の WCCP 設定情報の表示

いくつかの Content Engine CLI コマンドを使用して、スタンドアロン Content Engine 上の WCCP 関連設定情報（現在設定されている WCCP サービスのリストなど）を表示できます。

スタンドアロン Content Engine に現在設定されている WCCP サービスのリストを表示するには、**show wccp services EXEC** コマンドを入力します。

```
ContentEngine# show wccp services
Services configured on this Content Engine
  Web Cache
  Reverse Proxy
  RTSP
  WMT
  MMSU
  DNS
  FTP
  RTSPU
  HTTPS Cache
  WCCPv2 Service 90
  WCCPv2 Service 91
  WCCPv2 Service 92
  WCCPv2 Service 93
  WCCPv2 Service 94
  WCCPv2 Service 95
  WCCPv2 Service 96
  WCCPv2 Service 97
```

サポート対象の WCCP サービスのリストについては、[表 B-3](#) を参照してください。

現在設定されている WCCP サービスについて、Content Engine のリストを表示するには、**show wccp content-engines EXEC** コマンドを入力します。

スタンドアロン Content Engine に設定された WCCP サービスのルータ関連情報を表示するには、**show wccp EXEC** コマンドを入力します。

スタンドアロン Content Engine の WCCP フロー リダイレクションのディセーブル化および再イネーブル化

デフォルトでは、WCCP フロー リダイレクションはスタンドアロン Content Engine 上でイネーブルです。

スタンドアロン Content Engine で WCCP フロー リダイレクションを再度イネーブルにするには、**wccp flow-redirect enable** グローバル コンフィギュレーション コマンドを使用します。

```
ContentEngine(config)# wccp flow-redirect enable
```

WCCP フロー リダイレクションをディセーブルにする場合は、このコマンドの **no** 形式を使用します。

```
ContentEngine(config)# no wccp flow-redirect enable
```

詳細は、「[WCCP フロー保護の設定](#)」(p.15-10) を参照してください。

スタンドアロン Content Engine における WCCP のシャットダウン

reload または **no wccp version** コマンドを入力すると、不完全な TCP 接続を排除するために、Content Engine は WCCP を適正にシャットダウンします。Content Engine は、すべての接続のサービスが完了するか、または WCCP Version 2 の最大待ち時間 (**wccp shutdown max-wait** コマンドで指定 [デフォルトは 120 秒]) が経過するまで、再起動しません。

WCCP が適正にシャットダウンしている間、Content Engine は処理中のフローのサービスは継続しますが、新しいフローはバイパスします。フロー数が 0 になると、Content Engine はクラスタから除外され、その Content Engine のパケットは、リード Content Engine により他の Content Engine に再割り当てされます。Content Engine がクラッシュしたり、WCCP が適正にシャットダウンされずに再起動すると、TCP 接続が不完全になることがあります。

Content Engine 上の特定 ポートの WCCP サービス (ポート 80 のリバース プロキシ サービスなど) は、個別にシャットダウンすることはできないので、Content Engine の WCCP をシャットダウンする必要があります。Content Engine の WCCP をシャットダウンしても、Content Engine の WCCP 設定は保持され、プロキシ スタイル要求 (Content Engine がクライアントのブラウザから直接受信する HTTP 要求など) の処理は継続されます。

スタンドアロン Content Engine で WCCP をシャットダウンするには、Content Engine GUI または CLI を使用します。Content Engine GUI の場合、Content Engine GUI のメインウィンドウの **Clean WCCP shutdown** チェック ボックスにチェックマークを付けて (図 4-19 を参照)、同じウィンドウの **REBOOT** ボタンをクリックします。

Content Engine CLI を使用して、スタンドアロン Content Engine の WCCP を適正にシャットダウンする手順は、次のとおりです。

ステップ 1 適正なシャットダウンの待ち時間を指定します。

```
Content Engine(config)# wccp shutdown max-wait seconds
```

seconds に、**no wccp version** コマンドの入力後、Content Engine が WCCP の適正なシャットダウンを開始するまでの最大待ち時間を秒数 (0 ~ 86400) で指定します。デフォルトは 120 秒です。このコマンドがサポートされるのは、WCCP Version 2 だけです。次に、1000 秒の待ち時間を設定する例を示します。

```
ContentEngine(config)# wccp shutdown max-wait 1000
```

ステップ 2 Content Engine 上で WCCP Version 2 をシャットダウンします。

```
ContentEngine(config)# no wccp version 2
```

Content engine は 1000 秒後に WCCP Version2 のシャットダウンを開始します。何秒後に Content Engine の WCCP がシャットダウンされるかを示すカウントダウン メッセージが表示されます。

```
Waiting (999 seconds) for WCCP shutdown. Press ^C to skip shutdownn
```

カウントダウン メッセージが表示されたあと、**^C** を同時に押すと、シャットダウンが進行中でも中止することができます。

WCCP 透過リダイレクション用のルータの設定

ここでは、ルータ上での次の作業の方法について説明します。

- [WCCP Version 2 対応ルータのパスワードの設定 \(p.6-19\)](#)
- [ルータ上での一般的な WCCP Version 2 設定の実行 \(p.6-19\)](#)
- [ルータでの WCCP のイネーブル化 \(p.6-20\)](#)
- [WCCP サービス グループをサポートするための WCCP Version 2 ルータのイネーブル化 \(p.6-20\)](#)
- [WCCP を使用する発信 / 着信インターフェイスでのパケットリダイレクションのイネーブル化 \(p.6-24\)](#)
- [ルータ アクセス リストを使用した Content Engine のバイパス \(p.6-25\)](#)



(注)

ルータで実行する WCCP サービスを指定するには、WCCP Version 2 対応ルータ上で、`ip wccp` コマンドを使用します。WCCP サービスのサービス番号または名前指定します。サポート対象の WCCP サービス、サービス番号、および名前の完全なリストは、[表 B-3](#) を参照してください。ルータ上での WCCP サービスの設定方法については、「[ルータでの WCCP サービスの設定](#)」(p.6-27) を参照してください。

WCCP Version 2 対応ルータのパスワードの設定

スタンドアロン Content Engine がアクセスする WCCP Version 2 対応ルータのパスワードを、次のように設定する必要があります。

```
Router(config)# ip wccp web-cache password [0-7] password
```

各パラメータの内容：

- `password` は、指定されたサービス グループから受信するメッセージに MD5 認証を適用するように、WCCP Version 2 対応ルータに指示する文字列です。この認証によって許可されないメッセージは、廃棄されます。
- `0-7` は、パスワードの暗号化に使用される HMAC MD5 アルゴリズムを示すオプション値です。暗号化されたパスワードが Content Engine に対して作成されるときに、この値が生成されます。
- `password` は、ルータと Content Engine 間の接続に対するセキュリティを確立するために、HMAC MD5 値と組み合わせるオプションのパスワード名です。

ルータ上での一般的な WCCP Version 2 設定の実行

次に、ルータ上での一般的な WCCP Version 2 設定セッションの例を示します。



(注)

WCCP Version 2 を使用したリダイレクションをイネーブルにするには、すべての WCCP Version 2 のルータ設定で `ip wccp version 2` コマンドを入力する必要があります。

```
Router# configure terminal
Router(config)# ip wccp version 2
Router(config)# interface ethernet0
Router(config-if)# ip wccp web-cache redirect out
```

ルータでの WCCP のイネーブル化

ルータで WCCP をイネーブルにするには、`ip wccp version` グローバル コンフィギュレーション コマンドを入力します。たとえば、次のコマンドでルータの WCCP Version 2 をイネーブルにします。

```
Router(config)# ip wccp version 2
```



(注)

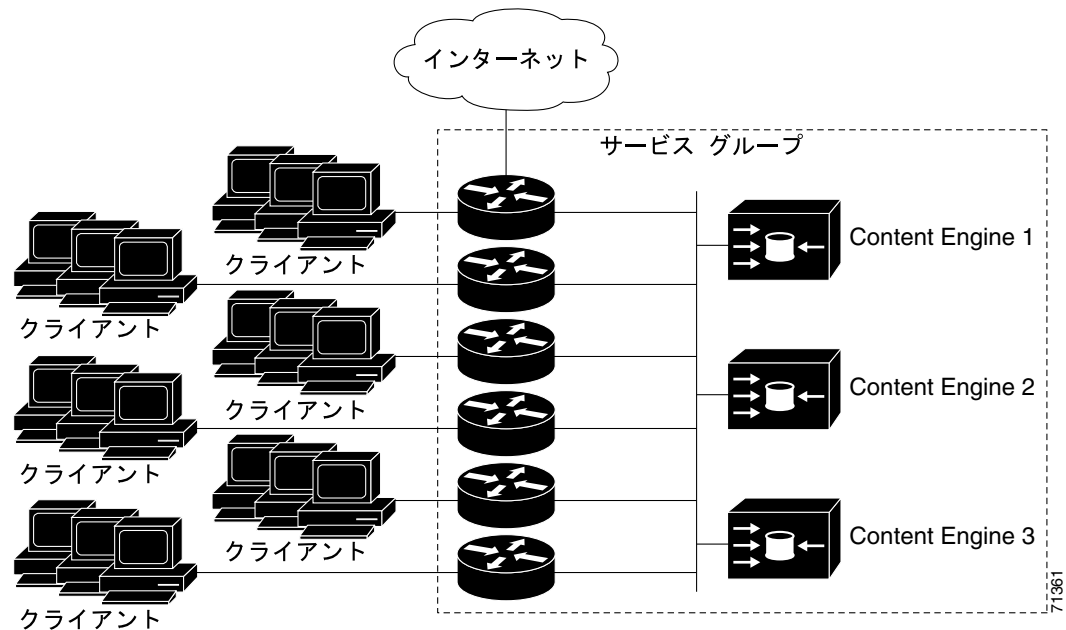
ルータリストに含まれる各ルータで WCCP をイネーブルにするためには、`ip wccp` グローバル コンフィギュレーション コマンドを使用する必要があります。

WCCP サービス グループをサポートするための WCCP Version 2 ルータのイネーブル化

WCCP Version 2 では、Content Engine クラスタの一連の Content Engine をイネーブルにして、複数のルータに接続します。同じサービスを実行しているクラスタ内の Content Engine および Content Engine に接続された WCCP Version 2 対応ルータを、サービス グループと呼びます。Content Engine のクラスタには、スタンドアロン Content Engine (Content Distribution Manager に登録されていない Content Engine) を含めることができます。

Content Engine との通信により、WCCP Version 2 対応ルータは使用可能な Content Engine を認識します。ルータと Content Engine は相互に認識しあい、WCCP Version 2 を使用するサービス グループを構築します。図 6-2 を参照してください。

図 6-2 WCCP Version 2 のサービス グループ



71361

サービス グループが確立されると、いずれかの Content Engine が、Content Engine クラスタの Content Engine 間の負荷の割り当てを決定するように指定されます。サポートされる WCCP サービスのタイプは、WCCP Version 1 または Version 2 のどちらを使用するかによって異なります。標準 web キャッシュ サービス (サービス 0) を除き、表 B-3 にリストされたすべての WCCP サービスでは、WCCP Version 2 が必要になります。

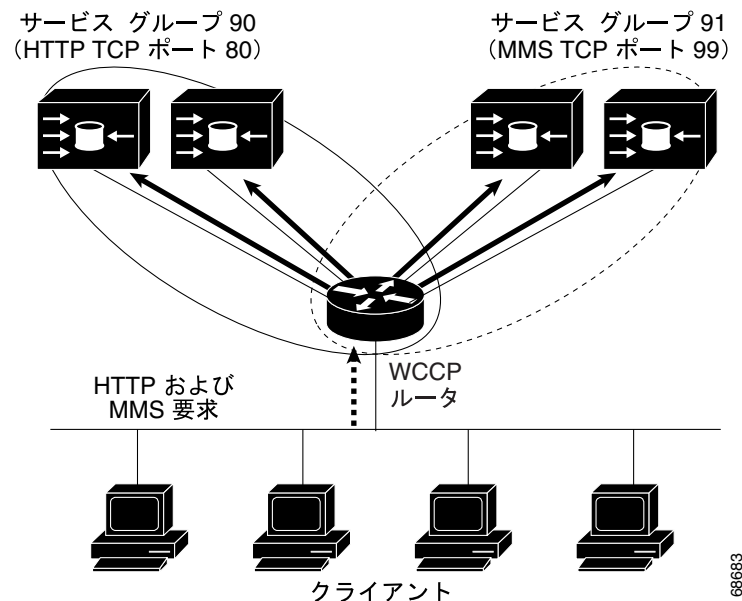
WCCP は、サービス グループの概念を使用して、WCCP Version 2 対応ルータおよびクラスタ内の Content Engine に、キャッシング関連サービスを定義します。また WCCP は、キャッシング関連サービスを要求するクライアントからのユーザ要求を、リアルタイムでクラスタにリダイレクトします。WCCP を使用した透過キャッシングでは、WCCP Version 2 対応ルータを設定して、透過キャッシング エンジンとして機能している Content Engine に要求をリダイレクトできます。

同じ WCCP サービス グループのメンバーとして設定され、リダイレクトされたトラフィックを受信するポートはすべて、次の特性を共有します。

- **wccp service-number mask** グローバル コンフィギュレーション コマンドによって設定された同じハッシュ パラメータまたはマスク パラメータを共有
- 各ポート上の WCCP Version 2 サービスは、個別に停止または開始できない (WCCP Version 2 の制約事項)

図 6-3 では、左側の 2 つの Content Engine は、ポート 80 で HTTP トラフィックだけを処理し、サービス グループ 90 のメンバーとして定義されています。右側の 2 つの Content Engine は、ポート 99 で Microsoft Media Server (MMS) 要求だけを処理し、サービス グループ 91 のメンバーとして定義されています。

図 6-3 WCCP Version 2 サービス グループ



カスタム Web キャッシュ サービスおよびリバース プロキシ サービス (サービス 98 と 99) は、それぞれ 1 ポートだけを使用して設定できます。したがって、従来のサービスを 1 つだけ設定している場合、透過リダイレクション ポートの最大合計数は 57 になります。これらの両方の従来型サービスが設定される場合、最大合計ポート数は 50 になります。

8 つのユーザ定義のサービスがそれぞれ最大ポート数の 8 ポートを使用すると、透過リダイレクションで指定可能なポートの最大数は 64 になります。

WCCP は、非対称パケットフローも処理できます。また WCCP は、WCCP サービス グループで使われるスイッチまたはルータの数（最大 32 のルータまたはスイッチが、クラスタ内で最大 32 の Content Engine と通信）に関係なく、Web サーバとキャッシュとの一貫したマッピングを常に保持します。

WCCP Version 2 対応ルータで WCCP サービス グループのサポートをイネーブルまたはディセーブルにするには、**ip wccp** グローバル コンフィギュレーション コマンドを使用します。WCCP サービス グループのサポート機能をディセーブルにするには、このコマンドの **no** 形式を使用します。

ip wccp {web-cache | service-number} [group-address groupaddress] [redirect-list access-list] [group-list access-list] [password [0-7] password]

表 6-4 に、**ip wccp** コマンドのパラメータを示します。

表 6-4 ip wccp コマンドのパラメータ

web-cache	web キャッシュ サービスをイネーブルにします。
<i>service-number</i>	WCCP Version 2 対応ルータが制御する WCCP サービスの識別番号。サービス番号は 0 ~ 99 です。WCCP のサービス番号のリストは、表 B-3 を参照してください。
group-address	(任意) WCCP Version 2 対応ルータが、指定したマルチキャスト IP アドレスを使用して WCCP サービス グループと通信するように設定します。
<i>groupaddress</i>	(任意) リダイレクトされたメッセージを受信する Content Engine を判別するために、WCCP Version 2 対応ルータが使用するマルチキャストアドレス。
redirect-list	(任意) WCCP Version 2 対応ルータが、アクセス リストを使用して、WCCP サービス グループにリダイレクトするトラフィックを制御するように設定します。
<i>access-list</i>	(任意) Content Engine にリダイレクトされるトラフィックを判別するアクセス リストの名前 (64 文字以下の文字列)。
group-list	(任意) WCCP Version 2 対応ルータが、アクセス リストを使用して、WCCP サービス グループへの参加が許可されている Content Engine を判別するように設定します。
<i>access-list</i>	(任意) WCCP サービス グループへの参加が許可されている Content Engine を判別するアクセス リストの名前 (64 文字以下の文字列)。
password	(任意) 指定されたサービス グループから受信したメッセージに MD5 認証を適用するように、WCCP Version 2 対応ルータに指示する文字列。この認証によって許可されないメッセージは、廃棄されます。
<i>0-7</i>	(任意) パスワードの暗号化に使用される HMAC MD5 アルゴリズムを示す値。暗号化されたパスワードが Content Engine に対して作成されるときに、この値が生成されます。
<i>password</i>	(任意) WCCP Version 2 対応ルータと Content Engine 間の接続に対するセキュリティを確立するために、HMAC MD5 値と組み合わせるパスワード名。

次に、マルチキャストアドレス 172.31.0.0 を使用 (待ち受け) して、WCCP リバース プロキシ サービス (サービス 99) を実行するように WCCP Version 2 対応ルータを設定する例を示します。

```
Router(config)# ip wccp 99 group-address 172.31.0.0
```



(注) WCCP Version 2 対応ルータ上のインターフェイスを設定し、WCCP 機能用の IP マルチキャストパケットの受信をイネーブルまたはディセーブルにするには、**ip wccp group-listen** コマンドを使用します。

WCCP Version 2 は、WCCP サービス グループに含める WCCP Version 2 対応ルータと Content Engine を制御できる認証機能を備えています。サービス グループのメンバーシップを管理するには、**ip wccp password [0-7] password** コマンドで設定するパスワードと HMAC MD5 アルゴリズムを使用します。

ルータでの WCCP リダイレクションのイネーブル化

WCCP Version 2 を使用する前に、インターネットに接続しているインターフェイス上、および Content Engine に接続しているインターフェイス上で、IP を設定する必要があります。Content Engine に接続するインターフェイスは、イーサネット インターフェイス、またはファスト イーサネット インターフェイスです。

ルータ インターフェイス上で WCCP Version 2 をイネーブルにし、ACNS 5.x ソフトウェアを実行しているスタンドアロン Content Engine に Web トラフィックをリダイレクトする手順は、次のとおりです。

ステップ 1 ルータで WCCP を使用できるようにします。

```
Router# configure terminal  
Router(config)# ip wccp version 2
```

ステップ 2 (任意) リダイレクト アクセス リストを指定します。

このアクセス リストと一致するパケットだけが、Content Engine にリダイレクトされます。リダイレクト アクセス リストを指定しない場合、すべてのパケットが Content Engine にリダイレクトされます。

```
Router(config)# ip wccp redirect-list [number | name]
```

ステップ 3 インターフェイス名および番号を指定し、インターフェイス コンフィギュレーション モードを開始します。

次に、Ethernet 0 インターフェイスを指定する例を示します。

```
Router(config)# interface ethernet 0
```

ステップ 4 インターネットに接続しているルータ インターフェイスを設定し、Web トラフィックをスタンドアロン Content Engine にリダイレクトします。

```
Router(config-if)# ip wccp web-cache redirect [in | out]
```



(注) すべての WCCP 対応ルータは **out** オプションをサポートしていますが、**in** オプションをサポートするのは特定のルータだけです。したがって、可能な場合には、**out** オプションを指定することを推奨します。詳細は、次項の「[WCCP を使用する発信 / 着信インターフェイスでのパケット リダイレクションのイネーブル化](#)」を参照してください。

■ WCCP 透過リダイレクション用のルータの設定

- ステップ 5** (任意) クライアントと Content Engine が同じネットワーク上にある場合、ルータがインターフェイス上で高速スイッチングパスを使用するように設定します。

```
Router(config-if)# ip route-cache same-interface
```

- ステップ 6** コンフィギュレーションモードを終了します。

```
Router(config-if)# end
```

- ステップ 7** 実行コンフィギュレーションをスタートアップ コンフィギュレーションに保存し、不揮発性メモリに保管します。

```
Router # copy running-config startup-config
```

WCCP を使用する発信 / 着信インターフェイスでのパケットリダイレクションのイネーブル化

リダイレクションは、発信インターフェイスまたは着信インターフェイスのいずれかに設定できます。着信トラフィックは、Cisco Express Forwarding (CEF)、Distributed Cisco Express Forwarding (dCEF)、Fast Forwarding、または Process Forwarding を使用するように設定できます。

インターフェイス上で着信トラフィックをリダイレクトする WCCP を設定すると、発信トラフィックの CEF 転送に関連するオーバーヘッドを回避できます。発信インターフェイス上でリダイレクションをイネーブルにすると、すべてのインターフェイスに到達するすべてのパケットに低速スイッチングパスが適用されます。着信インターフェイス上でリダイレクションをイネーブルにすると、そのインターフェイスに到達するパケットだけが設定された機能パスを使用し、他のインターフェイスに到達するパケットは、より高速のデフォルトパスを使用します。ただし、すべてのルータが着信インターフェイスのリダイレクションをサポートしているわけではありません。

また、着信トラフィックに対して WCCP を設定すると、ルーティングテーブルの検索前にパケットが分類されるので、パケットのリダイレクションは高速になります。

WCCP を使用する発信または着信インターフェイス上でパケットのリダイレクションをイネーブルにするには、表 6-5 に示すように、`ip wccp redirect` インターフェイス コンフィギュレーション コマンドを使用します。WCCP リダイレクションをディセーブルにする場合は、このコマンドの `no` 形式を使用します。

```
ip wccp {web-cache | service-number} redirect {out | in}
```

表 6-5 ip wccp redirect コマンドのパラメータ

パラメータ	説明
<code>web-cache</code>	web キャッシュ サービスをイネーブルにします。
<code>service-number</code>	WCCP Version 2 対応ルータによって制御される WCCP サービス グループの識別番号。サービス番号は 0 ~ 99 です。サポートされるサービス番号のリストは、表 B-3 を参照してください。
<code>redirect</code>	発信または着信インターフェイス上でパケットリダイレクションチェックをイネーブルにします。
<code>out</code>	発信インターフェイス上でパケットのリダイレクションを指定します。
<code>in</code>	着信インターフェイス上でパケットのリダイレクションを指定します。

ip wccp redirect インターフェイス コマンドは、**ip wccp redirect exclude in** コマンドに影響を与える可能性があります。インターフェイス上に **ip wccp redirect exclude in** を設定したあとで **ip wccp redirect in** コマンドを設定すると、**exclude in** コマンドはディセーブルになります。逆の場合も同様です。**exclude in** コマンドを設定すると、**redirect in** コマンドはディセーブルになります。

特定のクライアントのキャッシングをディセーブルにする手順は、次のとおりです。

ステップ 1 リダイレクションをイネーブルにするために使用するアクセス リストを設定します。

```
Router# configure terminal
Router(config)# ip wccp web-cache redirect-list access-list number
```

ステップ 2 Content Engine へのトラフィックのリダイレクションをイネーブルまたはディセーブルにするアクセス リストを作成します。

```
Router(config)# access-list access-list number deny host host-address
```

ステップ 3 任意ホストへのアクセスをイネーブルにするアクセス リストを設定します。

```
Router(config)# access-list access-list number permit ip any
```

次に、Ethernet 0 インターフェイス上でリバース プロキシ パケットのリダイレクションをチェックし、Content Engine にリダイレクトする設定セッションの例を示します。

```
Router# configure terminal
Router(config)# ip wccp 99
Router(config)# interface ethernet 0
Router(config-if)# ip wccp 99 redirect out
```

次に、Ethernet 0/1 インターフェイスに着信する HTTP トラフィックを Content Engine にリダイレクトする設定セッションの例を示します。

```
Router# configure terminal
Router(config)# ip wccp web-cache
Router(config)# interface ethernet 0/1
Router(config-if)# ip wccp web-cache redirect in
```

ルータ アクセス リストを使用した Content Engine のバイパス

デフォルトでは、すべての HTTP パケットが Content Engine にリダイレクトされます。WCCP Version 2 対応ルータにアクセス リストを設定すると、スタンドアロン Content Engine へのトラフィックのリダイレクションを許可または拒否できます。次に、以下の条件に適合するトラフィックを、ルータから Content Engine にリダイレクトしないように設定する例を示します。

- ホスト 10.1.1.1 から発信され、他の任意のホストを宛先とするトラフィック
- 任意のホストから発信され、ホスト 10.255.1.1 を宛先とするトラフィック

```
Router# configure terminal
Router(config)# ip wccp web-cache redirect-list 120
Router(config)# access-list 120 deny ip host 10.1.1.1 any
Router(config)# access-list 120 deny ip any host 10.255.1.1
Router(config)# access-list 120 permit ip any
```

明示的に許可されていないトラフィックは、暗黙的にリダイレクションが拒否されます。**access-list 120 permit ip any** コマンドは、(任意の送信元から任意の宛先への)すべてのトラフィックの Content Engine へのリダイレクションを明示的に許可します。条件の照合はコマンドを入力した順序で行われるので、**permit** グローバル コマンドは最後に入力します。アクセス リストの詳細については、Cisco IOS ソフトウェアのマニュアルを参照してください。

アクセス リストと一致するパケットだけをリダイレクトするには、**ip wccp redirect-list** グローバル コンフィギュレーション コマンドを使用します。このコマンドは、Content Engine にリダイレクトするパケットを指定する場合に使用します。

WCCP をイネーブルにし、**ip wccp redirect-list** コマンドを使用しない場合には、すべての Web 関連パケットが Content Engine にリダイレクトされます。**ip wccp redirect-list** コマンドを指定すると、アクセス リストに一致するパケットだけがリダイレクトされます。

WCCP を使用して Content Engine への要求のリダイレクションを開始するときに必要なコマンドは、**ip wccp** グローバル コンフィギュレーション コマンドと **ip web-cache redirect** インターフェイス コンフィギュレーション コマンドだけです。WCCP 対応ルータのインターフェイスで発信パケットをチェックし、適切なパケットを Content Engine にリダイレクトするには、**ip web-cache redirect** インターフェイス コンフィギュレーション コマンドを使用します。**ip wccp** コマンドがイネーブルであっても、**ip web-cache redirect** コマンドがディセーブルの場合には、WCCP 対応ルータは Content Engine を認識するだけで、使用しません。

一部の Web サイトでは、認証にパケットの送信元 IP アドレスを使用します。Content Engine は、Web サイトへの要求の送信時に独自の IP アドレスを使用します。したがって、Content Engine からの要求は認証されないことがあります。このような場合には、**ip wccp redirect-list** コマンドを使用して、Content Engine をバイパスします。

ip wccp redirect-list {number | name}

各パラメータの内容：

- *number* は、標準または拡張 IP アクセス リスト番号 (1 ~ 199) です。
- *name* は、標準または拡張 IP アクセス リストの名前です。この引数を使用できるのは、Cisco IOS リリース 11.2 P だけです。

ルータでの WCCP サービスの設定

ここでは、WCCP Version 2 を実行しているルータ上での、次の WCCP サービスの設定方法について説明します。

- ルータでの標準 Web キャッシュ サービス (サービス 0) の設定 (p.6-27)
- ルータでの DNS キャッシング サービス (サービス 53) の設定 (p.6-28)
- ルータでの FTP-Native キャッシング サービス (サービス 60) の設定 (p.6-29)
- ルータでの HTTPS キャッシュ サービス (サービス 70) の設定 (p.6-30)
- ルータでの RTSP サービス (サービス 80) の設定 (p.6-30)
- ルータでの MMST サービス (サービス 81) の設定 (p.6-31)
- ルータでの MMSU サービス (サービス 82) の設定 (p.6-32)
- ルータでの WMT-RTSPU サービス (サービス 83) の設定 (p.6-33)
- ルータでのユーザ定義 WCCP サービス (サービス 90 ~ 97) の設定 (p.6-34)
- ルータでのカスタム Web キャッシュ サービス (サービス 98) の設定 (p.6-34)
- ルータでのリバース プロキシ サービス (サービス 99) の設定 (p.6-35)

ルータ上で WCCP サービスを設定したあと、リダイレクトされた要求を受け入れるよう Content Engine を設定する必要があることに注意してください。「WCCP 透過リダイレクションのスタンドアロン Content Engine の設定」(p.6-8)を参照してください。

ルータでの標準 Web キャッシュ サービス (サービス 0) の設定

標準 web キャッシュ サービス (サービス 0) は、1 台の WCCP Version 1 対応ルータまたは 1 台または複数の WCCP Version 2 対応ルータを使用して、HTTP トラフィックをスタンドアロン Content Engine のポート 80 だけにリダイレクトするように事前定義された Web キャッシング サービスです。スタンドアロン Content Engine のポート 80 でリダイレクトされた HTTP 要求を受け入れるには、(透過 HTTP フォワード プロキシ キャッシュとして動作する) Content Engine に、このサービスを設定する必要があります。

次に、ルータで WCCP Version 2 をイネーブルにし、ルータに標準 web キャッシュ サービス (サービス 0) を設定する例を示します。

ステップ 1 ルータ上で WCCP Version 2 をイネーブルにします。

```
Router# configure terminal
Router(config)# ip wccp version 2
```

ステップ 2 ルータ上で標準 web キャッシュ サービス (サービス 0) をイネーブルにします。

```
Router(config)# ip wccp web-cache
```

ステップ 3 標準 web キャッシュ サービスを実行するインターフェイスを指定します。通常、このインターフェイスはインターネットに発信されるトラフィックを伝送します。

```
Router(config)# interface type number
```

次に、ルータの Ethernet インターフェイス 0/1 に標準 web キャッシュ サービスの実行を設定する例を示します。

```
Router(config)# interface ethernet 0/1
```

- ステップ 4** 標準 web キャッシュ サービスが設定されたインターフェイス (Ethernet インターフェイス 0/1 など) に着信する HTTP トラフィックを、ルータが確認するように設定します。ルータはこのトラフィックを確認して、パケットをスタンドアロン Content Engine にリダイレクトする必要があるかどうかを判別します。Content Engine は、WCCP Version 2 対応ルータからリダイレクトされた HTTP 要求をポート 80 で受け入れる透過フォワードプロキシサーバとして動作しています。

```
Router(config-if)# ip wccp web-cache redirect out
```

スタンドアロン Content Engine のポート 80 でリダイレクトされた HTTP 要求を受け入れるには、Content Engine 上にも標準 web キャッシュ サービス (透過 HTTP フォワードプロキシキャッシング) を設定する必要があることに注意してください。詳細は、「[スタンドアロン Content Engine での標準 Web キャッシュ サービス \(サービス 0\) の設定](#)」(p.7-18) を参照してください。

ルータでの DNS キャッシング サービス (サービス 53) の設定

DNS キャッシング サービス (サービス 53) は、事前定義された WCCP Version 2 キャッシング サービスです。このサービスは、Content Engine が DNS 名を解決できるように、WCCP Version 2 対応ルータがクライアント要求を Content Engine に透過的にリダイレクトすることを許可します。Content Engine は、DNS 名を解決すると、以降の DNS 要求で使用できるように、解決した名前をローカルに保存します。

ルータで DNS キャッシング サービス (サービス 53) を設定する手順は、次のとおりです。

- ステップ 1** ルータ上で WCCP Version 2 をイネーブルにします。

```
Router# configure terminal
Router(config)# ip wccp version 2
```

- ステップ 2** ルータ上で DNS キャッシング サービス (サービス 53) をイネーブルにします。

```
Router(config)# ip wccp 53
```

- ステップ 3** DNS キャッシング サービスを実行するルータ インターフェイスを指定します。

```
Router(config)# interface type number
```

- ステップ 4** ルータが、DNS キャッシング サービスに発信インターフェイスを使用するように設定します。

```
Router(config-if)# ip wccp 53 redirect out
```

Content Engine で WCCP Version 2 対応ルータからリダイレクトされた DNS 要求を受け入れるには、スタンドアロン Content Engine にも DNS キャッシング サービス（サービス 53）を設定する必要があります。詳細は、「スタンドアロン Content Engine での DNS キャッシングの設定」(p.7-67) を参照してください。

ルータでの FTP-Native キャッシング サービス（サービス 60）の設定

ftp-native キャッシング サービス（サービス 60）は、事前定義された WCCP Version 2 キャッシング サービスです。このサービスは、WCCP Version 2 対応ルータが、Content Engine 上の単一ポートに FTP ネイティブ要求を透過的にリダイレクトすることを許可します。Content Engine は、要求された FTP コンテンツを取得し、ローカルにコピーを保存し（ネイティブ FTP キャッシング）、要求されたコンテンツを FTP クライアントに配信します。



(注) ACNS ソフトウェア リリース 5.3.1 から、FTP ネイティブ要求と FTP-over-HTTP 要求を明確に区別するために、このサービスの名前が「ftp」から「ftp-native」に変更されました。サービス 60 (ftp-native キャッシング サービス) は、FTP ネイティブ要求の透過リダイレクションだけに適用され、FTP-over-HTTP 要求には適用されません。

ルータに ftp-native キャッシング サービス（サービス 60）を設定する手順は、次のとおりです。

ステップ 1 ルータ上で WCCP Version 2 をイネーブルにします。

```
Router# configure terminal  
Router(config)# ip wccp version 2
```

ステップ 2 ルータ上で ftp-native キャッシング サービス（サービス 60）をイネーブルにします。

```
Router(config)# ip wccp 60
```

ステップ 3 ftp-native キャッシング サービスを実行するインターフェイスを指定します。

```
Router(config)# interface type number
```

ステップ 4 ルータが、ftp-native キャッシング サービスに発信インターフェイスを使用するように設定します。

```
Router(config-if)# ip wccp 60 redirect out
```

Content Engine で WCCP Version 2 対応ルータからリダイレクトされた FTP ネイティブ要求を受け入れるには、スタンドアロン Content Engine にも ftp-native キャッシング サービスを設定する必要があります。詳細は、「透過 FTP ネイティブ キャッシングの設定」(p.7-56) を参照してください。

ルータでの HTTPS キャッシュ サービス（サービス 70）の設定

https-cache サービス（サービス 70）は、事前定義された WCCP Version 2 Web キャッシング サービスです。このサービスは、WCCP Version 2 対応ルータが、スタンドアロン Content Engine のポート 443 に、HTTPS トラフィックを透過的にリダイレクトすることを許可します。

ルータに https-cache サービス（サービス 70）を設定する手順は、次のとおりです。

ステップ 1 ルータ上で WCCP Version 2 をイネーブルにします。

```
Router# configure terminal
Router(config)# ip wccp version 2
```

ステップ 2 ルータ上で https-cache サービス（サービス 70）をイネーブルにします。

```
Router(config)# ip wccp 70
```

ステップ 3 https-cache サービスを実行するインターフェイスを指定します。

```
Router(config)# interface type number
```

次に、ルータの Ethernet 0 インターフェイスに https-cache サービスの実行を設定する例を示します。

```
Router(config)# interface ethernet 0
```

ステップ 4 ルータが、https-cache サービスに発信インターフェイスを使用するように設定します。

```
Router(config-if)# ip wccp 70 redirect out
```

Content Engine で WCCP Version 2 対応ルータからリダイレクトされた HTTPS 要求を受け入れるには、スタンドアロン Content Engine にも HTTPS 透過キャッシングを設定する必要があることに注意してください。詳細は、「[スタンドアロン Content Engine での HTTPS 透過キャッシングの設定](#)」(p.7-29) を参照してください。

ルータでの RTSP サービス（サービス 80）の設定

rtsp サービス（サービス 80）は、事前定義された WCCP Version 2 メディア キャッシング サービスです。このメディア キャッシング サービスは、WCCP Version 2 対応ルータが、Content Engine 上の単一ポートに RTSP クライアント要求を透過的にリダイレクトすることを許可します (RealMedia 透過キャッシング)。

Content Engine は、標準 RTSP ポート（デフォルト ポート 554）で、リダイレクトされた RTSP 要求を待ち受けます。デフォルト ポート（ポート 554）以外のポートで RTSP トラフィックを代行受信するには、ユーザ定義 WCCP サービス（サービス 90～97）を設定します。RealMedia クライアントからの RTSP 要求を透過的に代行受信する場合、必要な作業は、WCCP Version 2 対応ルータでの rtsp サービス（サービス 80）の設定だけです。

これに対し、WMT RTSP 要求を透過的に代行受信するには、WCCP Version 2 対応ルータに rtsp サービス（サービス 80）と wmt-rtspu サービス（サービス 83）の両方を設定する必要があります。ルータでのサービス 83 の設定の詳細については、「[ルータでの WMT-RTSPU サービス（サービス 83）](#)」

の設定」(p.6-33)を参照してください。

ルータに rtsp サービス (サービス 80) を設定する手順は、次のとおりです。

ステップ 1 ルータ上で WCCP Version 2 をイネーブルにします。

```
Router# configure terminal
Router(config)# ip wccp version 2
```

ステップ 2 ルータ上で rtsp サービス (サービス 80) をイネーブルにします。

```
Router(config)# ip wccp 80
```

ステップ 3 rtsp サービスを実行するインターフェイスを指定します。

```
Router(config)# interface type number
```

ステップ 4 ルータが、rtsp サービスに発信インターフェイスを使用するように設定します。

```
Router(config-if)# ip wccp 80 redirect out
```

Content Engine で WCCP Version 2 対応ルータからリダイレクトされた RealMedia RTSP 要求を受け入れるには、スタンドアロン Content Engine にも RealMedia 透過キャッシングを設定する必要があります。詳細は、第8章「スタンドアロン Content Engine の RealMedia サービスの設定」を参照してください。

ルータでの MMST サービス (サービス 81) の設定

MMST は、TCP を使用して転送を行う Microsoft Media Server プロトコルです。mmst サービス (サービス 81) は、事前定義された WCCP Version 2 メディア キャッシング サービスです。このサービスは、WCCP Version 2 対応ルータが MMST リダイレクションを使用して Content Engine の単一ポート (ポート 1755) に WMT クライアント要求を透過的にリダイレクトすることを許可します (Content Engine は、WMT 透過キャッシング用に設定された透過プロキシ サーバとして動作します)。

WMT MMS 要求を透過的に代行受信するには、WCCP Version 2 対応ルータに mmsu サービス (サービス 82) と mmst サービス (サービス 81) の両方を設定する必要があります。ルータでのサービス 82 の設定の詳細については、「ルータでの MMSU サービス (サービス 82) の設定」(p.6-32)を参照してください。

ルータに mmst サービス (サービス 81) を設定する手順は、次のとおりです。

ステップ 1 ルータ上で WCCP Version 2 をイネーブルにします。

```
Router# configure terminal
Router(config)# ip wccp version 2
```

■ ルータでの WCCP サービスの設定

ステップ 2 ルータ上で mmst サービス（サービス 81）をイネーブルにします。

```
Router(config)# ip wccp 81
```

ステップ 3 mmst サービスを実行するインターフェイスを指定します。

```
Router(config)# interface type number
```

ステップ 4 ルータが、mmst サービスに発信インターフェイスを使用するように設定します。

```
Router(config-if)# ip wccp 81 redirect out
```

Content Engine で WCCP Version 2 対応ルータからリダイレクトされた WMT 要求を受け入れるには、スタンドアロン Content Engine に WMT MMS 透過キャッシングを設定する必要があることに注意してください。詳細は、「[WMT 要求の MMS 透過リダイレクションの設定](#)」(p.9-43) を参照してください。

ルータでの MMSU サービス（サービス 82）の設定

MMSU は、UDP を使用して転送を行う Microsoft Media Server プロトコルです。mmsu サービス（サービス 82）は、WCCP Version 2 メディア キャッシング サービスです。このサービスは、WCCP Version 2 ルータが MMSU リダイレクションを使用して Content Engine の単一ポート（ポート 1755）に WMT クライアント要求を透過的にリダイレクトすることを許可します（Content Engine は、WMT MMS 透過キャッシング用に設定された透過プロキシ サーバになります）。

WMT MMS 要求を透過的に代行受信するには、WCCP Version 2 対応ルータに mmst サービス（サービス 81）と mmsu サービス（サービス 82）の両方を設定する必要があります。ルータでのサービス 82 の設定の詳細については、「[ルータでの MMST サービス（サービス 81）の設定](#)」(p.6-31) を参照してください。

ルータに mmsu サービス（サービス 82）を設定する手順は、次のとおりです。

ステップ 1 ルータ上で WCCP Version 2 をイネーブルにします。

```
Router# configure terminal
Router(config)# ip wccp version 2
```

ステップ 2 ルータ上で mmsu サービス（サービス 82）をイネーブルにします。

```
Router(config)# ip wccp 82
```

ステップ 3 mmsu サービスを実行するインターフェイスを指定します。

```
Router(config)# interface type number
```

ステップ 4 ルータが、mmsu サービスに発信インターフェイスを使用するように設定します。

```
Router(config-if)# ip wccp 82 redirect out
```

Content Engine で WCCP Version 2 対応ルータからリダイレクトされた WMT MMS 要求を受け入れるには、スタンドアロン Content Engine に WMT MMS 透過キャッシングを設定する必要があることに注意してください。詳細は、「[WMT 要求の MMS 透過リダイレクションの設定](#)」(p.9-43) を参照してください。

ルータでの WMT-RTSPU サービス (サービス 83) の設定

wmt-rtspu サービス (サービス 83) は、事前定義された WCCP Version 2 メディア キャッシング サービスです。このサービスは、WCCP Version 2 対応ルータが Content Engine の単一ポートに、Windows Media 9 プレーヤーからの RTSP クライアント要求を透過的にリダイレクトすることを許可します (Content Engine は、WMT RTSP 透過キャッシング用に設定された透過プロキシ サーバとして動作します)。wmt-rtspu サービスは ACNS ソフトウェア リリース 5.3.1 で追加された機能で、rtspu サービスとも呼ばれています。

Content Engine は、標準 RTSPU ポート (デフォルト ポート 5005) で、リダイレクトされた RTSP 要求を待ち受けます。デフォルト ポート (ポート 5005) 以外のポートで WMT RTSP トラフィックを代行受信するには、ユーザ定義 WCCP Version 2 サービス (サービス 90 ~ 97) を設定します。

ルータに wmt-rtspu サービス (サービス 83) を設定する手順は、次のとおりです。

ステップ 1 ルータ上で WCCP Version 2 をイネーブルにします。

```
Router# configure terminal
Router(config)# ip wccp version 2
```

ステップ 2 ルータ上で wmt-rtspu サービス (サービス 83) をイネーブルにします。

```
Router(config)# ip wccp 83
```

ステップ 3 wmt-rtspu サービスを実行するインターフェイスを指定します。

```
Router(config)# interface type number
```

ステップ 4 ルータが、wmt-rtspu サービスに発信インターフェイスを使用するように設定します。

```
Router(config-if)# ip wccp 82 redirect out
```

Content Engine への WMT RTSP 要求の透過リダイレクションをサポートするには、WCCP ルータにも rtsp サービス (サービス 80) を設定する必要があります。

Content Engine で WCCP Version 2 対応ルータからリダイレクトされた WMT RTSP 要求を受け入れるには、スタンドアロン Content Engine にも WMT RTSP 透過キャッシングを設定する必要があることに注意してください。詳細は、「[スタンドアロン Content Engine での WMT RTSP ストリーミング およびキャッシング サービスの設定](#)」(p.9-21) を参照してください。

ルータでのユーザ定義 WCCP サービス（サービス 90 ～ 97）の設定

ルータが WCCP Version 2 を使用してユーザ定義 WCCP サービス（サービス 90 ～ 97）をサポートするように設定する手順は、次のとおりです。

ステップ 1 ルータ上で WCCP Version 2 をイネーブルにします。

```
Router# configure terminal
Router(config)# ip wccp version 2
```

ステップ 2 ユーザ定義サービス（サービス 90 など）用の WCCP 機能をイネーブルにします。

```
Router(config)# ip wccp 90
```

ステップ 3 ルータでサービス 90 を実行するインターフェイスを指定します。

```
Router(config)# interface type number
```

ステップ 4 Content Engine が、サービス 90 に発信インターフェイスを使用するように設定します。

```
Router(config-if)# ip wccp 90 redirect out
```

Content Engine で WCCP Version 2 対応ルータからリダイレクトされたプロキシ パケットを受け入れるには、スタンドアロン Content Engine にもユーザ定義 WCCP サービス（サービス 90 など）を設定する必要もあることに注意してください。詳細は、「[ユーザ定義 WCCP サービスをサポートするためのスタンドアロン Content Engine の設定](#)」(p.6-14) を参照してください。

ルータでのカスタム Web キャッシュ サービス（サービス 98）の設定

custom-web キャッシュ サービス（サービス 98）は、事前定義された WCCP Version 2 Web キャッシング サービスです。このサービスは、WCCP Version 2 対応ルータが Content Engine のポート 80 以外の複数のポートに HTTP トラフィックをリダイレクトすることを許可します。Content Engine は、透過フォワードプロキシサーバとして動作します。この WCCP サービスにより、ユーザ定義 WCCP サービス（サービス 90 ～ 97）を設定しなくても、Content Engine の複数のポート（最大 8 ポート）で、WCCP によりリダイレクトされた HTTP 要求を待ち受けるように Content Engine を設定できます。

WCCP Version 2 を使用して custom-web キャッシュ サービス（サービス 98）をサポートできるようにルータを設定する手順は、次のとおりです。

ステップ 1 ルータ上で WCCP Version 2 をイネーブルにします。

```
Router# configure terminal
Router(config)# ip wccp version 2
```

ステップ 2 ルータ上で custom-web キャッシュ サービス（サービス 98）をイネーブルにします。

```
Router(config)# ip wccp 98
```

ステップ 3 custom-web キャッシュ サービスを実行するインターフェイスを指定します。

```
Router(config)# interface type number
```

次に、Ethernet 0 インターフェイスに custom-web キャッシュ サービスの実行を設定する例を示します。

```
Router(config)# interface ethernet 0
```

ステップ 4 ルータが、custom-web キャッシュ サービスに発信インターフェイスを使用するように設定します。

```
Router(config-if)# ip wccp 98 redirect out
```

Content Engine の複数ポートで WCCP Version 2 対応ルータからリダイレクトされた Web キャッシュ プロキシ パケットを受け入れるには、スタンドアロン Content Engine にも custom-web キャッシュ サービスを設定する必要があることに注意してください。詳細は、「[スタンドアロン Content Engine でのカスタム Web キャッシュ サービス \(サービス 98\) の設定](#)」(p.7-20) を参照してください。

ルータでのリバース プロキシ サービス (サービス 99) の設定

reverse-proxy サービス (サービス 99) は、事前定義された WCCP Version 2 サービスです。このサービスは、WCCP Version 2 対応ルータが、透過リバース プロキシ サーバとして動作しているスタンドアロン Content Engine にリバース プロキシ パケットをリダイレクトすることを許可します。

WCCP Version 2 を使用して reverse-proxy サービス (サービス 99) をサポートできるようにルータを設定する手順は、次のとおりです。

ステップ 1 ルータ上で WCCP Version 2 をイネーブルにします。

```
Router# configure terminal
Router(config)# ip wccp version 2
```

ステップ 2 ルータ上で reverse-proxy サービス (サービス 99) をイネーブルにします。

```
Router(config)# ip wccp 99
```

ステップ 3 reverse-proxy サービスを実行するインターフェイスを指定します。

```
Router(config)# interface type number
```

次に、Ethernet 0 インターフェイスに reverse-proxy サービスの実行を設定する例を示します。

```
Router(config)# interface ethernet 0
```

- ステップ 4** ルータが、reverse-proxy サービスに発信インターフェイスを使用するように設定します。ルータは、Ethernet 0 インターフェイス上でリバース プロキシ パケットを確認し、(透過リバース プロキシ サーバとして動作している) Content Engine に透過的にリダイレクトする必要があるパケットかどうかを判別します。

```
Router(config-if)# ip wccp 99 redirect out
```

Content Engine で WCCP Version 2 ルータからリダイレクトされた リバース プロキシ パケットを受け入れるには、スタンドアロン Content Engine にも reverse-proxy サービス (サービス 99) を設定する必要があることに注意してください。詳細は、「[スタンドアロン Content Engine での HTTP リバース プロキシ キャッシングの設定](#)」(p.7-23) を参照してください。

ルータでの WCCP 統計情報の削除

特定のサービス、またはすべてのサービスについて、WCCP Version 2 ルータ上の WCCP 統計情報を削除するには、`clear ip wccp EXEC` コマンドを使用します。

```
clear ip wccp {web-cache | service-number}
```

各パラメータの内容：

- `web-cache` で、ルータから web キャッシュ サービスの統計情報を削除することを指定します。
- `service-number` で、ルータから統計情報を削除する特定サービスの番号を指定します。サービスグループ番号は、0～99 です。

たとえば、ルータから reverse-proxy サービス（サービス 99）の統計を削除するには、WCCP Version 2 対応ルータに次のコマンドを入力します。

```
Router# clear ip wccp web-cache 99
```

WCCP レイヤ 2 サポートの設定

ルータまたはスイッチ上の WCCP は、レイヤ 2 のハードウェアで、WCCP のトラフィック代行受信およびリダイレクト機能を部分的または完全に実現するスイッチング ハードウェアを利用できます。これにより、Content Engine は、互換性のある Cisco スイッチに直接接続されている場合、レイヤ 2 または MAC アドレスのリライトリダイレクションを実行できます。このリダイレクション方式は、スイッチング ハードウェアにより加速されるので、GRE を使用したレイヤ 3 リダイレクションよりも効率的です。

Content Engine には、スイッチとのレイヤ 2 接続が必要です。スイッチと Content Engine 間の GRE トンネルには要件がないので、スイッチは、CLI で `l2-redirect` オプションを指定することにより、カットスルー方式でカプセル化パケットを転送できます。

レイヤ 2 転送を選択する場合、WCCP Version 2 対応ルータまたはスイッチと Content Engine 間で、2 つの負荷分散方式を使用できます。

- ハッシュ アサイメント

Catalyst 6000 および 6500 シリーズ スイッチの場合、この負荷分散方式は、WCCP レイヤ 2 Policy Feature Card (PFC) リダイレクションと呼ばれます。この方式は、Supervisor Engine 1A と Multilayer Switch Feature Card 2 (MSFC2) を組み合わせて使用し、最高 3 Gbps (ギガビット / 秒) の転送パフォーマンスを実現します。

- マスク アサイメント

この負荷分散方式は、WCCP レイヤ 2 Policy Feature Card 2 (PFC2) リダイレクションと呼ばれます。この方式では、Supervisor Engine 2 と MSFC2 を組み合わせて使用します。

Content Engine GUI または CLI コマンドを使用して、Content Engine 上で特定の WCCP サービスの負荷分散方式を指定できます。Content Engine のサポート対象の WCCP サービスはすべて、`wccp custom-web-cache`、`wccp media-cache`、`wccp reverse-proxy`、`wccp service-number`、`wccp web-cache`、`wccp wmt`、`wccp rtsp` などのコマンドを使用して、レイヤ 2 転送でのハッシュアサイメントまたはマスクアサイメントのいずれかの負荷分散方式をサポートできます。Content Engine クラスタの WCCP サービスごとに、1 つの負荷分散方式（ハッシュおよびマスク）を指定できます。たとえば、Content Engine クラスタ A に 3 つの WCCP サービスを定義した場合、クラスタ A の 2 つのサービスにハッシュ負荷分散方式を設定します。クラスタ A の 3 つめのサービスには、マスク負荷分散方式を設定します。



(注) レイヤ 2 リダイレクションのマスク アサイメント負荷分散方式を設定するには、Content Engine CLI を使用します (Content Engine GUI からは設定できません)。

ハッシュ負荷分散方式の設定方法の詳細は、「レイヤ 2 転送のハッシュ負荷分散方式の設定」を参照してください。マスク負荷分散方式の設定方法の詳細は、「レイヤ 2 転送のマスク負荷分散方式の設定」(p.6-39) を参照してください。

レイヤ 2 転送のハッシュ負荷分散方式の設定

2 つのタイプのパケット転送方式 (レイヤ GRE およびレイヤ 2 リダイレクション) は、いずれも負荷分散方式としてハッシュをサポートしています。ハッシュでは、リダイレクトされたトラフィックを、Content Engine クラスタの複数の Content Engine 間で負荷分散する方法を指定できます。

Content Engine でユーザ定義 WCCP Version 2 サービスを設定する場合、特定のユーザ定義 WCCP Version 2 サービスでのハッシュ パラメータ (送信元 IP アドレスでハッシュなど) を設定できます。ユーザ定義 WCCP サービス (サービス 90 ~ 97) のデフォルトのハッシュ アサイメントは、宛先 IP アドレスでのハッシュです。任意のユーザ定義 WCCP サービスについて、デフォルトのハッシュ アサイメントを変更するには、**wccp service-number** グローバル コンフィギュレーション コマンドを使用します。

次に、**wccp service-number** コマンドを使用して、ユーザ定義 WCCP サービス (この例ではサービス 90) に宛先 IP アドレスではなく送信元 IP アドレスでのハッシュを設定する例を示します。

```
ContentEngine(config)# wccp service-number 90 router-list-num 1 port-list-num 1
application cache hash-source-ip
```

表 6-6 に、デフォルトのハッシュ アサイメントを示します。事前定義された WCCP サービスのデフォルトのハッシュ アサイメントは、変更できないことに注意してください。

表 6-6 WCCP Version 2 サービスのデフォルトのハッシュ アサイメント

WCCP サービス タイプ	サービス番号 (ID)	サービスのデフォルトのハッシュ アサイメント
ユーザ定義 WCCP サービス	90 ~ 97	デフォルトのハッシュ アサイメントは宛先 IP アドレスでのハッシュ
事前定義された WCCP サービス		
web-cache	0	宛先 IP アドレスでのハッシュ
dns caching	53	送信元ポートでのハッシュ
ftp-native	60	宛先 IP アドレスでのハッシュ
https-cache	70	送信元 IP アドレスでのハッシュ
rtsp	80	宛先 IP アドレスでのハッシュ
mmst	81	宛先 IP アドレスでのハッシュ
mmsu	82	宛先 IP アドレスでのハッシュ
wmt-rtspu	83	宛先 IP アドレスでのハッシュ
custom-web-cache	98	宛先 IP アドレスでのハッシュ
reverse-proxy	99	送信元 IP アドレスと送信元ポートでのハッシュ

次に、Content Engine が、Multilayer Switch Feature Card (MSFC) と Supervisory Engine 1A (SUP 1A) を実装している Catalyst 6500 シリーズ スイッチから、ハッシュ アサイメント負荷分散を使用してレイヤ2 リダイレクトトラフィックを受信するように設定する例を示します。Content Engine を設定する手順は、次のとおりです。

ステップ 1 Content Engine 上で WCCP Version 2 をイネーブルにします。

```
ContentEngine# configure terminal
ContentEngine(config)# wccp version 2
```

ステップ 2 Content Engine でルータ リストを作成します。次に、単一の WCCP Version 2 対応ルータ (IP アドレス 172.16.55.1 のルータ) だけを含む、ルータ リスト 1 を作成する例を示します。

```
ContentEngine(config)# wccp router-list 1 172.16.55.1
```

ステップ 3 Content Engine 上で標準 web キャッシュ サービス (サービス 0) を設定します。

ステップ 2 で作成したルータ リストを使用して、この WCCP サービスを設定します。**l2-redirect** オプションを入力して、このサービスのパケット転送方式として (GRE ではなく) レイヤ2 リダイレクションを指定します。マスク アサイメント方式は指定しないので、リダイレクトされた要求は、デフォルトのハッシュ アサイメント方式により負荷分散されます。

```
ContentEngine(config)# wccp web-cache router-list-num 1 l2-redirect
```

ステップ 4 **show wccp services detail EXEC** コマンドを使用して設定を表示し、内容を確認します。

```
ContentEngine# show wccp services detail
```

ステップ 5 実行コンフィギュレーションを不揮発性メモリに書き込みます。

```
ContentEngine# copy running-config startup-config
```

レイヤ 2 転送のマスク負荷分散方式の設定

2つのタイプのパケット転送方式 (GRE およびレイヤ2 リダイレクション) は、いずれもマスク負荷分散方式をサポートしています。Content Engine 上で異なるマスク (宛先 IP マスクなど) を指定するには、**wccp service-name mask** グローバル コンフィギュレーション コマンドを使用します。

特定の WCCP Version 2 サービスのデフォルト マスクを変更するには、**wccp service-name mask** グローバル コンフィギュレーション コマンドを使用します。たとえば、透過的にリダイレクトされる HTTPS 要求 (https-cache サービス) のマスクを設定するには、**wccp https-cache mask** グローバル コンフィギュレーション コマンドを使用します。

```
ContentEngine(config)# wccp https-cache mask ?
dst-ip-mask      Specify sub-mask used in packet destination-IP address
dst-port-mask    Specify sub-mask used in packet destination-port number
src-ip-mask      Specify sub-mask used in packet source-IP address
src-port-mask    Specify sub-mask used in packet source-port number
```

```
wccp https-cache {mask {[dst-ip-mask hex_num] [dst-port-mask port_hex_num]
[src-ip-mask hex_num] [src-port-mask port_hex_num]}}
```

表 6-7 では、コマンドのパラメータを説明しています。

表 6-7 wccp https-cache CLI コマンドのパラメータ

パラメータ	説明
mask	Content Engine の割り当てに使用するマスク。最低 1 つのマスクを設定します。最大 4 つのマスクを設定できます。
dst-ip-mask	(任意) リダイレクトされたパケットの宛先 IP アドレスとの照合に使用するマスク。
<i>hex_num</i>	16 進数で定義された IP アドレス マスク (0xFC000000 など)。範囲は、0x00000000 ~ FC000000 です。
dst-port-mask	(任意) リダイレクトされたパケットの宛先ポート番号との照合に使用するマスク。
<i>port_hex_num</i>	16 進数で定義された送信元ポート マスク (0xFC00 など)。ポート範囲は 0 ~ 65535 です。
src-ip-mask	(任意) リダイレクトされたパケットの送信元 IP アドレスとの照合に使用するマスク。
<i>hex_num</i>	16 進数で定義された IP アドレス マスク (0xFC000000 など)。範囲は、0x00000000 ~ FC000000 です。
src-port-mask	(任意) リダイレクトされたパケットの送信元ポート番号との照合に使用するマスク。
<i>port_hex_num</i>	16 進数で定義された送信元ポート マスク (0xFC00 など)。ポート範囲は 0 ~ 65535 です。

特定の WCCP Version 2 サービスのマスク設定を表示するには、**show wccp masks service-name EXEC** コマンドを使用します。

```
ContentEngine(config)# show wccp masks ?
custom-web-cache Custom web caching service
dns              DNS caching service
ftp-native       Native FTP caching service
https-cache      HTTPS caching service
reverse-proxy    Reverse Proxy web caching service
rtsp            Media caching service
service-number   Custom-service number
web-cache        Standard web caching service
wmt             WMT service
wmt-rtspu       WMT RTSPU service
```


次に、Content Engine が、Multilayer Switch Feature Card 2 と Supervisory Engine 2 (MSFC2/SUP 2) を実装している Catalyst 6500 シリーズ スイッチから、レイヤ 2 でリダイレクトされたトラフィックを受信するように設定する例を示します。Content Engine を設定する手順は、次のとおりです。

ステップ 1 Content Engine 上で WCCP Version 2 をイネーブルにします。

```
ContentEngine# configure terminal
ContentEngine(config)# wccp version 2
```

ステップ 2 Content Engine でルータ リストを作成します。次に、単一の WCCP Version 2 対応ルータ (IP アドレス 172.16.55.1) だけを含む、ルータ リスト 1 を作成する例を示します。

```
ContentEngine(config)# wccp router-list 1 172.16.55.1
```

ステップ 3 Content Engine 上で web キャッシュ サービスを設定します。ステップ 2 で作成されたルータ リストを使用してこの WCCP サービスを設定します。**l2-redirect** オプションを入力して、パケット転送方式として (GRE ではなく) レイヤ 2 リダイレクションを指定します。**mask-assign** オプションを入力して、この WCCP サービスの負荷分散方式として (デフォルトのハッシュ アサイメント方式ではなく) マスク アサイメントを指定します。

```
ContentEngine(config)# wccp web-cache router-list-num 1 l2-redirect mask-assign
```

ステップ 4 設定を表示して、内容を確認します。

```
ContentEngine# show wccp services detail
```

ステップ 5 実行コンフィギュレーションを不揮発性メモリに書き込みます。

```
ContentEngine# copy running-config startup-config
```

スタンドアロン Content Engine での WCCP サービスの設定例

ここでは、スタンドアロン Content Engine での WCCP Version 2 を使用した WCCP サービスの設定について、次の例を紹介します。

- 例 1 — WCCP Version 1 を使用した Web キャッシュ サービスの設定 (p.6-43)
- 例 2 — WCCP Version 2 を使用した Web キャッシュ サービスの設定 (p.6-46)
- 例 3 — WCCP Version 2 を使用した HTTPS 透過キャッシング サービスの設定 (p.6-48)
- 例 4 — スタンドアロン Content Engine での複数の WCCP Version 2 サービスの設定 (p.6-49)

WCCP Version 1 または WCCP Version 2 のどちらかを使用して、標準 web キャッシュ サービス(サービス 0) を設定する場合、次の重要事項に注意してください。

- Content Engine のパケットは暗号化または圧縮されず、「内部」の NAT ファイアウォール (存在する場合) の一部である必要があります。
- Content Engine と WCCP 対応ルータをファイアウォールで隔てることはできません。ファイアウォールは、オリジン Web サーバ宛てのパケット トラフィックだけを処理し、サーバの代わりに Content Engine によりクライアントに送信されるパケット トラフィックは処理されないからです。
- Web キャッシュ リダイレクト対応インターフェイスを越えて、サーバへのルート上に Content Engine を配置すると、IP ルート キャッシュ にエントリが読み込まれません。
- Content Engine の標準 web キャッシュ サービスの設定には、Content Engine GUI も使用できません。ただし、ルータに標準 web キャッシュ サービスを設定する場合には、常に CLI を使用する必要があります。
- WCCP 対応ルータを使用して Web キャッシュ サービスをサポートするには、インターネットに接続しているインターフェイス上に IP アドレスを設定し、そのインターフェイスを Content Engine に接続する必要があります。IP が設定されたルータ上のインターフェイスが現在、使用可能かどうかを確認するには、ルータで **show ip interface EXEC** コマンドを使用します。

インターフェイスが使用可能である場合、Cisco IOS ソフトウェアは、ルーティング テーブルに直接接続ルートを自動的に入力します。使用可能なインターフェイスとは、Cisco IOS ソフトウェアがパケットの送受信時に使用できるインターフェイスです。Cisco IOS ソフトウェアは、インターフェイスが使用不可である場合、直接接続されているルーティングのエントリをルーティング テーブルから削除します。Cisco IOS ソフトウェアは、このエントリを削除すると、ダイナミック ルーティング プロトコルを使用して、ネットワークへのバックアップ ルート (ある場合) を判別できます。

インターフェイスで双方向通信が可能な場合、回線プロトコルに「up」と表示されます。インターフェイス ハードウェアが使用可能である場合、インターフェイスに「up」と表示されます。

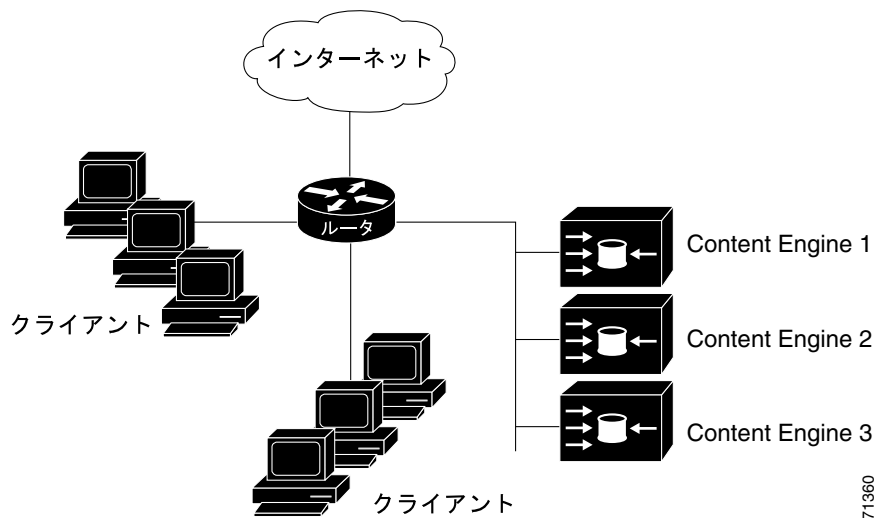
- オプションのインターフェイス タイプを指定すると、特定のインターフェイスだけの情報が表示されます。
- オプションの引数を指定しない場合は、すべてのインターフェイスについての情報が表示されます。

非同期インターフェイスが PPP (ポイントツーポイント プロトコル)、または Serial Line Internet Protocol (SLIP; シリアル回線インターネット プロトコル) によってカプセル化されている場合、IP 高速スイッチングがイネーブルです。PPP または SLIP によりカプセル化される非同期インターフェイスについて、**show ip interface** コマンドを入力すると、IP 高速スイッチングがイネーブルであることを示すメッセージが表示されます。

例 1 — WCCP Version 1 を使用した Web キャッシュ サービスの設定

WCCP Version 1 を使用して標準 web キャッシュ サービス（サービス 0）を実行するには、単一の WCCP 対応ルータと 1 台または複数の Content Engine を設定します。図 6-4 に、3 台の Content Engine クラスタと、これらの Content Engine を処理する単一の WCCP Version 1 対応ルータを含む、WCCP Version 1 ネットワークの構成例を示します。

図 6-4 WCCP Version 1 を使用した Content Engine のネットワーク構成



71360



(注)

WCCP Version 1 では、単一の WCCP 対応ルータが 1 つの Content Engine のクラスタを処理し、そのルータがクラスタのデフォルト ホームルータになります。WCCP Version 1 では、クラスタにサービスを提供する単一ルータが、すべての IP パケットのリダイレクションを実行します。

WCCP Version 1 を使用する場合、Content Engine と、これらの Content Engine にサービスする単一 WCCP 対応ルータ（ホームルータ）の間で、次のイベントシーケンスが発生します。

1. 各 Content Engine が、Content Engine クラスタと通信する WCCP 対応ルータの IP アドレスを記録します。
2. Content Engine が自身の IP アドレスを WCCP 対応ルータに送信し、クラスタ内の Content Engine が相互に存在を通知します。
3. WCCP 対応ルータが Content Engine に応答し、各 Content Engine がクラスタ内の他の Content Engine に接続できるようにし、クラスタ内の Content Engine アドレスのビュー（リスト）を提供して、すべての Content Engine が互いに認識できることを示します。
4. このビューが確立されると、1 台の Content Engine がリードエンジンとして指定され、IP パケットのリダイレクト方法を WCCP 対応ルータに指示します。

リード Content Engine は、クラスタで番号が最小の IP アドレスを持つものとして定義され、クラスタを処理する WCCP Version 1 対応ルータ（ホームルータ）から参照されます。

■ スタンドアロン Content Engine での WCCP サービスの設定例

次に、Content Engine CLI を使用して、WCCP Version 1 を使用した標準 web キャッシュ サービス (サービス 0) をイネーブルにし、設定する例を説明します。WCCP Version 1 では、1 つのサービス (標準 web キャッシュ サービス) と単一 WCCP 対応ルータ (ホーム ルータ) だけを設定できます。Content Engine のクラスタがある場合でも、単一の WCCP Version 1 対応ルータだけが Content Engine のクラスタと通信し、クラスタのデフォルト ホーム ルータになります。

この例では、IP アクセス リストを使用して、スタンドアロン Content Engine にリダイレクトする Web 関連パケットを制御しています。また、単一ルータおよびスタンドアロン Content Engine に設定した web キャッシュ サービスが正常に動作しているかどうかを確認する例も示します。

ステップ 1 WCCP 対応ルータを使用して標準 web キャッシュ サービスを実行するには、インターネットに接続しているインターフェイスに IP アドレスを設定し、そのインターフェイスをスタンドアロン Content Engine に接続する必要があります。

ルータに **show ip interface EXEC** コマンドを入力し、IP が設定されたルータ上のインターフェイスが使用可能かどうかを確認します。

ステップ 2 **show ip wccp EXEC** コマンドを入力して、WCCP がルータで現在、イネーブルに設定されているかどうかを確認します。

ステップ 3 ルータの WCCP Version 1 をイネーブルにし、ルータが WCCP Version 1 を使用して、宛先が 192.168.196.51 以外の Web 関連パケットをスタンドアロン Content Engine にリダイレクトするように設定します。

```
Router# configure terminal
Router(config)# ip wccp version 1
Router(config)# access-list 100 deny ip any host 192.168.196.51
Router(config)# access-list 100 permit ip any any
Router(config)# ip wccp web-cache redirect-list 100
Router(config)# interface Ethernet 0
Router(config)# ip wccp web-cache redirect out
Router(config-if)# end
Router#
%SYS-5-CONFIG_I:Configured from console by console.
```

ステップ 4 スタンドアロン Content Engine 上で WCCP Version 1 をイネーブルにします。

```
ContentEngine# configure terminal
ContentEngine(config)# wccp version 1
```

ステップ 5 **wccp home-router ip-address** グローバル コンフィギュレーション コマンドを入力して、スタンドアロン Content Engine にホーム ルータを指定します。IP デフォルト ゲートウェイの IP アドレスを指定することもできます。

次に、IP アドレスが 192.168.51.102 のホーム ルータを指定する例を示します。

```
ContentEngine(config)# wccp home-router 192.168.51.102
```



(注) Content Engine GUI (WCCP > **Enable WCCP** を選択) を使用して、スタンドアロン Content Engine に WCCP Version 1 の web キャッシュ サービスを設定することもできます。Enable WCCP ウィンドウの詳細については、ウィンドウの **HELP** ボタンをクリックしてください。

ステップ6 Content Engine 上で標準 web キャッシュ サービスを設定します。詳細は、「[スタンドアロン Content Engine での標準 Web キャッシュ サービス \(サービス 0\) の設定](#)」(p.7-18) を参照してください。

ステップ7 `show wccp EXEC` コマンドを入力して、スタンドアロン Content Engine 上で標準 web-cache サービスがイネーブルになっていることを確認します。

```
ContentEngine# show wccp services
Services configured on this Content Engine
      Web Cache
ContentEngine#
```

Content Engine 上で `show wccp EXEC` コマンドの他のオプションを指定すると、スタンドアロン Content Engine の他の WCCP 情報を表示できます。たとえば、Content Engine の WCCP GRE パケット関連情報を表示するには、`show wccp` コマンドの `gre` オプションを指定します。

ステップ8 WCCP 対応ルータ上で `show ip wccp web-cache` コマンドを入力し、ルータの WCCP がイネーブルで、Web キャッシュとして設定したスタンドアロン Content Engine をルータが認識していることを確認します。

次の例では、ホーム ルータの設定直後に `show ip wccp web-caches` コマンドを入力しています。数秒後、2 番目の出力にあるように、IP アドレス 192.168.25.3 のスタンドアロン Content Engine のステータスが、「NOT Usable」から「Usable」に変更されます。

```
Router# show ip wccp web-cache

WCCP Web-Cache information:
  IP Address:          192.168.25.3
  Protocol Version:   1.0
  State:               NOT Usable
  Initial Hash Info:  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
                    FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
  Assigned Hash Info: 000000000000000000000000000000000000
                    00000000000000000000000000000000
  Hash Allotment:     0 (0.00%)
  Packets Redirected: 0
  Connect Time:       00:00:06

Router# show ip wccp web-cache

WCCP Web-Cache information:
  IP Address          192.168.25.3
  Protocol Version:   0.3
  State:              Usable
  Initial Hash Info:  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
                    FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
  Assigned Hash Info: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
                    FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
  Hash Allotment:     256 (100.00%)
  Packets Redirected: 0
  Connect Time:       00:00:31
```

ステップ9 WCCP Version 1 対応ルータ上で `debug ip wccp events EXEC` コマンドを入力し、重要な WCCP イベントの情報を表示します。

次に、WCCP Version 1 対応ルータで使用可能な Web キャッシュのリストに 1 台の Content Engine が追加された場合の `debug ip wccp events` コマンドの出力例を示します。

```
Router# debug ip wccp events

WCCP-EVNT: Built I_See_You msg body w/1 usable web caches, change # 0000000A
WCCP-EVNT: Web Cache 192.168.25.3 added
WCCP-EVNT: Built I_See_You msg body w/2 usable web caches, change # 0000000B
WCCP-EVNT: Built I_See_You msg body w/2 usable web caches, change # 0000000C
```

ステップ 10 WCCP Version 1 対応ルータ上で `debug ip wccp packets` コマンドを入力し、このルータで送受信された各 WCCP パケットの情報を表示します。

次に、`debug ip wccp packets` コマンドの出力例を示します。このルータは、192.168.25.3 のスタンドアロン Content Engine に、キープアライブパケットを送信しています。各キープアライブパケットには、ID 番号が対応付けられています。Content Engine は、ルータからキープアライブパケットを受信すると、ID 番号を付けた応答をルータに戻します。

```
Router# debug ip wccp packets
WCCP-PKT: Received valid Here_I_Am packet from 192.168.25.3 w/rcvd_id 00003532
WCCP-PKT: Sending I_See_You packet to 192.168.25.3 w/rcvd_id 00003534
WCCP-PKT: Received valid Here_I_Am packet from 192.168.25.3 w/rcvd_id 00003533
WCCP-PKT: Sending I_See_You packet to 192.168.25.3 w/rcvd_id 00003535
WCCP-PKT: Received valid Here_I_Am packet from 192.168.25.3 w/rcvd_id 00003534
WCCP-PKT: Sending I_See_You packet to 192.168.25.3 w/rcvd_id 00003536
WCCP-PKT: Received valid Here_I_Am packet from 192.168.25.3 w/rcvd_id 00003535
WCCP-PKT: Sending I_See_You packet to 192.168.25.3 w/rcvd_id 00003537
WCCP-PKT: Received valid Here_I_Am packet from 192.168.25.3 w/rcvd_id 00003536
WCCP-PKT: Sending I_See_You packet to 192.168.25.3 w/rcvd_id 00003538
WCCP-PKT: Received valid Here_I_Am packet from 192.168.25.3 w/rcvd_id 00003537
WCCP-PKT: Sending I_See_You packet to 192.168.25.3 w/rcvd_id 00003539
```



ヒント ルータ上の WCCP によってリダイレクトされたパケットのカウンタをクリアするには、WCCP 対応ルータ上で `clear ip wccp EXEC` コマンドを入力します。

例 2 — WCCP Version 2 を使用した Web キャッシュ サービスの設定

次に、クライアントとスタンドアロン Content Engine が同じサブネット上にあり、WCCP Version 1 ではなく WCCP Version 2 が使用されている場合に、Content Engine CLI を使用して標準 web キャッシュ サービス（サービス 0）を設定する例を示します。

ステップ 1 ステップ 3 でルータ リストに追加する各ルータについて、WCCP Version 2 がイネーブルであることを確認します。

```
Router(config)# ip wccp version 2
```

ステップ 2 `ip wccp web-cache password` グローバル コンフィギュレーション コマンドを入力して、WCCP 対応ルータに標準 web キャッシュ サービスと、ルータのパスワードを設定します。

```
Router(config)# ip wccp web-cache [password [0-7] [password]]
```

各パラメータの内容：

- **password** は、指定したサービス グループから受信するメッセージに MD5 認証を適用するように、WCCP 対応ルータに指示します。この認証によって許可されないメッセージは、廃棄されます。
- **0-7** は、パスワードの暗号化に HMAC MD5 アルゴリズムを使用することを示すオプション値です。この値は、Content Engine 用の暗号化パスワードの作成時に生成されます。
- **password** は、WCCP 対応ルータと Content Engine 間にセキュアな接続を確立するために、HMAC MD5 値と組み合わせるオプションのパスワードです。

ステップ 3 標準 web キャッシュ サービス (WCCP Version 2 を使用した HTTP 透過キャッシング) を設定するスタンドアロン Content Engine 上で、ルータ リスト (route-list 1 など) を作成します。

```
ContentEngine(config)# wccp router-list 1 10.10.10.1
```

ステップ 4 Content Engine で標準 web キャッシュ サービスをイネーブルにし、Content Engine に対して特定のサービスをサポートする WCCP Version 2 対応ルータ (ルータ リスト) (Content Engine のポート 80 に HTTP 要求をリダイレクトするルータ) を指定します。

この WCCP サービスと作成したルータ リストを対応付けます。レイヤ 2 リダイレクション オプションを割り当てます。マスク アサイメント方式を指定しない場合、デフォルトの負荷分散方式はハッシュ アサイメント方式です。

```
ContentEngine(config)# wccp web-cache router-list-num 1 12-redirect
```



ヒント Content Engine GUI を使用して (Content Engine GUI から **WCCP > Web Cache** を選択)、スタンドアロン Content Engine に WCCP Version 2 web キャッシュ サービスを設定することもできます。Content Engine GUI を使用して Content Engine に WCCP Version 2 設定する場合には、次の Content Engine GUI ウィンドウで、各 WCCP サービスに対応するルータ リストを指定する必要があります。Web Cache ウィンドウ (**WCCP > Web Cache**)、Reverse Proxy ウィンドウ (**WCCP > Reverse Proxy**)、Custom Web Cache ウィンドウ (**WCCP > Custom Web Cache**)、および WCCP Services ウィンドウ (**WCCP > Services**)。

ステップ 5 Content Engine 上で WCCP Version 2 をイネーブルにします。

```
ContentEngine(config)# wccp version 2
```

ステップ 6 グローバル コンフィギュレーション モードを終了します。

```
ContentEngine(config)# exit
```

ステップ 7 実行コンフィギュレーションを Content Engine の不揮発性メモリに書き込みます。

```
ContentEngine# write memory
```


■ スタンドアロン Content Engine での WCCP サービスの設定例

ステップ 8 ルータに WCCP Version 2 の実行が設定され、ルータ上で WCCP がモニタされます。

```
Router# show ip wccp
```

または

```
Router# show ip wccp {web-cache | 90-99}
```

ステップ 9 WCCP 対応ルータに、ルータが特定のサービス グループ内で検出した Content Engine の情報を問い合わせます。この情報は、90～99 の範囲の WCCP サービスに対して表示できます。

```
Router# show ip wccp {web-cache | 90-99} detail
```

ステップ 10 `ip wccp direct` コマンドがインターフェイス上で設定されているかどうかを判別します。

```
Router# show ip interface
```

ステップ 11 特定の WCCP サービス グループ内で検出されたデバイス、また現在のルータが接続されている他のすべてのルータで認識できない Content Engine を表示します。この情報は、90～99 のサービス グループに対して表示できます。

```
Router# show ip wccp {web-cache | 90-99} view
```

例 3 — WCCP Version 2 を使用した HTTPS 透過キャッシング サービスの設定

次に、Content Engine CLI を使用して、スタンドアロン Content Engine にユーザ定義 WCCP Version 2 サービスおよび HTTPS 透過キャッシングを設定する例を示します。この例では、サービス番号 95 がユーザ定義サービスです。

ステップ 1 Content Engine が、デフォルトのポート以外のポートで、複数の WCCP Version 2 対応ルータにより透過的に代行受信およびリダイレクトされる HTTPS 要求を受け入れられるように、サービス 95 を設定します。

最初に、サービス 95 で使用する ルータ リストおよびポートリスト（サービス 95 用のポート リスト 1 など）を作成する必要があります。ポート リストには、WCCP Version 2 対応ルータが サービス 95 の WCCP リダイレクションをサポートするポート番号が含まれています。

```
ContentEngine# configure terminal
ContentEngine(config)# wccp service-number 95 router-list-num 1
ContentEngine(config)# wccp service-number 95 router-list-num 1 port-list-num 1
```

ステップ 2 サービス 95 を設定する場合、トラフィックを Content Engine 上のキャッシング アプリケーション、HTTP キャッシング アプリケーション、またはストリーミング アプリケーションのどれかにリダイレクトするのかを指定する必要があります。

```
ContentEngine(config)# wccp service-number 95 router-list-num 1 port-list-num 1
application ?
  cache      Direct traffic to the caching application
  https-cache Direct traffic to the HTTPS caching application
  streaming  Direct traffic to the streaming media application
```

- ステップ 3** Content Engine が、HTTPS アプリケーションにリダイレクトされたトラフィックを受け入れるように指定します。ルータ リスト番号 1 の WCCP Version 2 対応ルータは、HTTPS トラフィックを Content Engine の HTTPS キャッシング アプリケーションにリダイレクトします。Content Engine は、ポート リスト 1 に指定されたポートで、WCCP によりリダイレクトされる要求を待ち受けます。

```
ContentEngine(config)# wccp service-number 95 router-list-num 1 port-list-num 1
https-cache
```

ACNS ソフトウェア リリース 5.2.1 以降では、https-cache サービスの accept-all モードがサポートされています。accept-all モードは、HTTPS トラフィックのフィルタリングをサポートします。このモードは、従来の WCCP サービス（デフォルトですべての Web トラフィックを代行受信する web キャッシュ サービスなど）と同様に機能します。

デフォルトでは、Content Engine はすべての HTTPS トラフィックを受け入れます。

```
ContentEngine(config)# wccp https-cache ?
accept-all      Accept all HTTPS traffic by default
mask            Specify mask used for CE assignment
router-list-num  Router list number
```



- (注)** **wccp https-cache accept-all** グローバル コンフィギュレーション コマンドを使用した場合、HTTPS キャッシュ (HTTPS 透過 キャッシング用に設定されている Content Engine) は、accept all モード (すべての HTTPS トラフィックを代行受信) で動作します。それ以外の場合には、HTTPS キャッシュは、ACNS 5.1.x ソフトウェアのように accept only モードで動作します。

- ステップ 4** Content Engine 上で WCCP Version 2 をイネーブルにします。

```
ContentEngine(config)# wccp version 2
```

例 4 — スタンドアロン Content Engine での複数の WCCP Version 2 サービスの設定

次に、Content Engine CLI を使用して、スタンドアロン Content Engine に 16 の WCCP Version 2 サービスを設定する例を示します。

- ステップ 1** 16 の WCCP Version 2 サービス (8 つのユーザ定義サービスと 8 つの事前定義サービス) をサポートする WCCP Version 2 対応ルータをリストしたルータ リストを設定します。この例では、router list 1 に、単一のルータ (IP アドレス 10.1.202.1 の WCCP Version 2 対応ルータ) が設定されます。

```
ContentEngine(config)# wccp router-list 1 10.1.202.1
```

- ステップ 2** 8 つのポート リスト (ポート リスト番号 1 ~ 8) を設定します。

これらのポート リストで、特定の WCCP Version 2 対応ルータから着信するトラフィックを待ち受ける Content Engine 上のポート番号を指定します。これらのポート リストにより、Content Engine が複数のポート上で着信 WCCP 要求を待ち受けるように設定できます。デフォルトでは、Content Engine はポート 80 で着信トラフィックを待ち受けます。設定する 8 つのユーザ定義 WCCP

■ スタンドアロン Content Engine での WCCP サービスの設定例

Version 2 サービス（サービス 90～97）について、それぞれ1つのポートリストを作成します。各ポートリストに、最大8のポートを定義できます。この例では、各ポートリストに1つのポートを指定しています（たとえば、ポートリスト1にはポート32が含まれます）。

```
ContentEngine(config)# wccp port-list 1 32
ContentEngine(config)# wccp port-list 2 33
ContentEngine(config)# wccp port-list 3 34
ContentEngine(config)# wccp port-list 4 35
ContentEngine(config)# wccp port-list 5 36
ContentEngine(config)# wccp port-list 6 37
ContentEngine(config)# wccp port-list 7 38
ContentEngine(config)# wccp port-list 8 39
```

- ステップ3** Content Engine で標準 web キャッシュ サービス（サービス 0）をイネーブルにし、ルータ リスト 1 と、最初の事前定義 WCCP サービスとを関連付けます。

```
ContentEngine(config)# wccp web-cache router-list-num 1
```

Content Engine は、ルータ リスト 1 のルータからリダイレクトされる HTTPS 要求を、ポート 80 で待ち受けます。



(注) *HTTP 要求*という用語は、HTTP、FTP-over-HTTP、および HTTPS-over-HTTP のすべての要求を意味します。

- ステップ4** Content Engine で reverse-proxy キャッシング サービス（サービス 99）をイネーブルにして、ルータ リスト 1 と、2 番めの事前定義 WCCP Version 2 サービスとを関連付けます。

```
ContentEngine(config)# wccp reverse-proxy router-list-num 1
```

Content Engine は、ルータ リスト 1 のルータからリダイレクトされるリバースプロキシ要求を、ポート 80 で待ち受けます。

- ステップ5** Content Engine で custom-web キャッシュ サービス（サービス 98）をイネーブルにして、ルータ リスト 1 およびポート 31 と、3 番めの事前定義 WCCP Version 2 サービスとを対応付けます。

```
ContentEngine(config)# wccp custom-web-cache router-list-num 1 port 31
```

Content Engine は、ルータ リスト 1 の WCCP Version 2 対応ルータからリダイレクトされる HTTP 要求を、ポート 31 で待ち受けます。

- ステップ6** Content Engine で rtsp サービス（サービス 80）をイネーブルにして、ルータ リスト 1 と、4 番めの事前定義 WCCP Version 2 サービスとを対応付けます。

```
ContentEngine(config)# wccp rtsp router-list-num 1
```

Content Engine は、ルータ リスト 1 の WCCP Version 2 対応ルータからリダイレクトされる RTSP 要求を、標準 RTSP ポート（デフォルトのポート 554）で待ち受けます。

- ステップ 7** Content Engine で WMT MMS サービス（サービス 81 および 82）をイネーブルにして、ルータ リスト 1 と、5 番めの事前定義 WCCP Version 2 サービスとを対応付けます。

```
ContentEngine(config)# wccp wmt router-list-num 1
```

このコマンドを指定後、Content Engine は、ルータ リスト 1 の WCCP Version 2 対応ルータからリダイレクトされる WMT 要求を、デフォルト ポート（ポート 1755）で待ち受けます。

- ステップ 8** Content Engine で dns キャッシング サービス（サービス 53）をイネーブルにして、ルータ リスト 1 と、6 番めの事前定義 WCCP Version 2 サービスとを対応付けます。

```
ContentEngine(config)# wccp dns router-list-num 1
```

Content Engine は、ルータ リスト 1 の WCCP Version 2 対応ルータからリダイレクトされる DNS 要求を、ポート 80 で待ち受けます。

- ステップ 9** Content Engine で ftp-native キャッシング サービス（サービス 60）をイネーブルにして、ルータ リスト 1 と、7 番めの事前定義 WCCP Version 2 サービスとを対応付けます。

```
ContentEngine(config)# wccp ftp-native router-list-num 1
```

Content Engine は、ルータ リスト 1 の WCCP Version 2 対応ルータからリダイレクトされる FTP ネイティブ要求を、ポート 80 で待ち受けます。これは、ネイティブ FTP キャッシング用です（Content Engine が FTP-over-HTTP 要求をクライアントのブラウザから直接受信し、要求されたコンテンツをキャッシュする FTP-over-HTTP キャッシングとは異なります）。

- ステップ 10** Content Engine で https-caching サービス（サービス 70）をイネーブルにして、ルータ リスト 1 と、8 番めの事前定義 WCCP Version 2 サービスとを対応付けます。

```
ContentEngine(config)# wccp https-cache router-list-num 1
```

ACNS 5.1 ソフトウェアでは、WCCP モードの Content Engine 上で、SSL ターミネーションの対象となる HTTPS 要求だけがサポートされます。ACNS 5.1 ソフトウェアでは、特定のサイト（Content Engine でのサポートが明確に設定されている HTTPS オリジン サーバ）への HTTPS 要求だけが、WCCP モードでの SSL ターミネーションの対象になります。ACNS 5.1 ソフトウェアでは、サポートされることが明示的に設定されていない HTTPS サーバへの HTTPS 要求は、Content Engine によりバイパスされます。SSL ターミネーションの詳細は、「[HTTPS クライアント要求の SSL ターミネーションの概要](#)」（p.7-26）を参照してください。

ACNS 5.1.x ソフトウェアでは、https-cache サービスに対してサポートされるのは、1つの代行受信モード（accept-only モード）だけです。accept-only モードでは、次のように、Content Engine が、リダイレクトされた特定の HTTPS サーバ宛での要求だけを受け入れるように設定する必要があります。

```
ContentEngine(config)# wccp https-cache router-list-num 1
```

または

```
ContentEngine(config)# wccp service-number 95 router-list-num 1 port-list 1  
https-cache
```

上記の両方の例の場合、（https server グローバル コンフィギュレーション コマンドにより）Content Engine に HTTPS サーバが設定されている場合に限り、Content Engine はリダイレクトされた HTTPS トラフィックを受け入れます。

ACNS ソフトウェア リリース 5.2.1 以降では、要求された HTTPS サーバが Content Engine 上に設定されていれば、WCCP モードおよびマニュアル プロキシ モードで HTTPS 要求の SSL ターミネーションがサポートされます。それ以外の HTTPS トラフィックは、トンネリングされます。HTTPS 要求のトンネリングの詳細は、「[HTTPS クライアント要求のトンネリングの概要](#)」(p.7-27)を参照してください。要求された特定のコンテンツをキャッシュするには、Content Engine 上に HTTPS サーバの適正な証明書と鍵をインポートし、Content Engine がこれらのサーバをキャッシュするように設定する必要があります。スタンドアロン Content Engine の場合、Content Engine CLI を使用して設定します。詳細は、「[HTTPS キャッシングに使用する証明書と秘密鍵の設定](#)」(p.7-34)を参照してください。

- ステップ 11** Content Engine で最初のユーザ定義 WCCP サービス (サービス 90) をイネーブルにして、このサービスに、ルータ リスト 1 およびポート リスト 1 を対応付けます。**application cache** オプションを入力し、トラフィックが Content Engine のキャッシング アプリケーションにリダイレクトされるように指定します。

```
ContentEngine(config)# wccp service-number 90 router-list-num 1 port-list-num 1
application cache
```

Content Engine は、ルータ リスト 1 のルータからリダイレクトされる要求を、ポート リスト 1 のポート (ポート 32) で待ち受けます。



ヒント この例では、WCCP ルータから Content Engine 上の (ストリーミング アプリケーションではなく) キャッシング アプリケーションにトラフィックをリダイレクトするので、設定する各ユーザ定義 WCCP サービス (サービス 90 ~ 97) について、**application cache** オプションを指定する必要があります。

- ステップ 12** Content Engine で 2 番目のユーザ定義 WCCP サービス (サービス 91) をイネーブルにして、このサービスに、ルータ リスト 1 およびポート リスト 2 を対応付けます。

```
ContentEngine(config)# wccp service-number 91 router-list-num 1 port-list-num 2
application cache
```

Content Engine は、ルータ リスト 1 の WCCP Version 2 対応ルータからリダイレクトされる要求を、ポート リスト 2 のポート (ポート 33) で待ち受けます。

- ステップ 13** Content Engine で 3 番目のユーザ定義 WCCP サービス (サービス 92) をイネーブルにして、このサービスに、ルータ リスト 1 およびポート リスト 3 を対応付けます。

```
ContentEngine(config)# wccp service-number 92 router-list-num 1 port-list-num 3
application cache
```

Content Engine は、ルータ リスト 1 のルータからリダイレクトされる要求を、ポート リスト 3 のポート (ポート 34) で待ち受けます。

- ステップ 14** Content Engine で 4 番目のユーザ定義 WCCP サービス (サービス 93) をイネーブルにして、このサービスに、ルータ リスト 1 およびポート リスト 4 を対応付けます。

```
ContentEngine(config)# wccp service-number 93 router-list-num 1 port-list-num 4
application cache
```

Content Engine は、ルータ リスト 1 のルータからリダイレクトされる要求を、ポート リスト 4 のポート（ポート 35）で待ち受けます。

- ステップ 15** Content Engine で 5 番目のユーザ定義 WCCP サービス（サービス 94）をイネーブルにして、このサービスに、ルータ リスト 1 およびポート リスト 5 を対応付けます。

```
ContentEngine(config)# wccp service-number 94 router-list-num 1 port-list-num 5
application cache
```

Content Engine は、ルータ リスト 1 のルータからリダイレクトされる要求を、ポート リスト 5 のポート（ポート 36）で待ち受けます。

- ステップ 16** Content Engine で 6 番目のユーザ定義 WCCP サービス（サービス 95）をイネーブルにして、このサービスに、ルータ リスト 1 およびポート リスト 6 を対応付けます。

```
ContentEngine(config)# wccp service-number 95 router-list-num 1 port-list-num 6
application cache
```

Content Engine は、ルータ リスト 1 のルータからリダイレクトされる要求を、ポート リスト 6 のポート（ポート 37）で待ち受けます。

- ステップ 17** Content Engine で 7 番目のユーザ定義 WCCP サービス（サービス 96）をイネーブルにして、このサービスに、ルータ リスト 1 およびポート リスト 7 を対応付けます。

```
ContentEngine(config)# wccp service-number 96 router-list-num 1 port-list-num 7
application cache
```

Content Engine は、ルータ リスト 1 のルータからリダイレクトされる要求を、ポート リスト 7 のポート（ポート 38）で待ち受けます。

- ステップ 18** Content Engine で 8 番目のユーザ定義 WCCP サービス（サービス 97）をイネーブルにして、このサービスに、ルータ リスト 1 およびポート リスト 8 を対応付けます。

```
ContentEngine(config)# wccp service-number 97 router-list-num 1 port-list-num 8
application cache
```

Content Engine は、ルータ リスト 1 のルータからリダイレクトされる要求を、ポート リスト 8 のポート（ポート 39）で待ち受けます。

- ステップ 19** Content Engine 上で WCCP Version 2 をイネーブルにします。

```
ContentEngine(config)# wccp version 2
```

- ステップ 20** Content Engine 上で WCCP スロー スタート機能をディセーブルにします。

```
ContentEngine(config)# no wccp slow-start enable
```

WCCP スロー スタート機能の詳細は、「[WCCP スロー スタートの設定](#)」(p.15-11) を参照してください。

リダイレクション方式としてのレイヤ4スイッチングの設定

レイヤ4スイッチング (Content Services Switch [CSS] スイッチ) を使用してスタンドアロン Content Engine に要求を透過的にリダイレクトする場合の透過リダイレクションの設定では、次の重要事項に注意してください。

- CSS スイッチは、透過プロキシキャッシングのほか、リバースプロキシキャッシングもサポートします。CSS スイッチは、Content Engine へのデータの配信方法 (URL 全体、URL ストリング、ドメイン名全体、ドメイン文字列など) に応じて、複数の負荷分散方式をサポートしています。また CSS スイッチは、既知のキャッシュ可能オブジェクトのリストを作成します。このリストは変更できますが、変更作業の大半は、Content Engine のキャッシング機能によって軽減されます。
- CSS スイッチを設定して、コンテンツを動的に分析し、キャッシング可能なコンテンツかどうかを判別できます。キャッシング可能である場合、CSS スイッチはそのコンテンツをキャッシュ サービスに送信します。キャッシングできない場合、CSS スイッチはそのコンテンツをオリジン Web サーバに直接送信します。
- 透過キャッシュ構成のすべてのキャッシュ サーバが使用不可の場合、CSS スイッチですべてのクライアント要求をオリジン Web サーバに転送できます。

CSS スイッチを使用した透過キャッシングを設定するには、次の作業が必要です。

1. CSS スイッチ上での透過キャッシングのイネーブル化
2. CSS スイッチからリダイレクトされたトラフィックを受け入れるための、スタンドアロン Content Engine の設定
3. スタンドアロン Content Engine での透過キャッシングのイネーブル化

次の例では、CLI コマンドを使用して、CSS スイッチとスタンドアロン Content Engine を使用した透過キャッシングを設定する方法を示します。この例では、CS150 という名前の CSS スイッチと、CE100 という名前のスタンドアロン Content Engine を使用する透過キャッシング サービスとして、serv1 を設定しています。CSS スイッチを使用したキャッシングを設定する前に、インターフェイス、サービス、所有者、VLAN、およびコンテンツ ルールが設定されていることを確認してください。



(注)

CSS スイッチにこれらの属性を設定する方法については、『*Content Services Switch Basic Configuration Guide*』を参照してください。各コマンドの詳細については、『*Content Services Switch Command Reference*』を参照してください。

レイヤ4スイッチングを使用した透過キャッシングの設定例

次に、(WCCP 対応ルータではなく) レイヤ4 CSS スイッチとスタンドアロン Content Engine を使用した透過キャッシングをイネーブルにする例を示します。透過キャッシングをイネーブルにする手順は、次のとおりです。

- ステップ 1** CSS スイッチで、透過キャッシング用に予約されたサービス serv1 を追加します。

```
CS150(config)# add service serv1
CS150(config-service[serv1])#
```

- ステップ 2** serv1 のサービスタイプとして、透過キャッシングを指定します。

```
CS150(config-service[serv1])# type transparent cache
```


- ステップ 3** CSS スイッチでキャッシュするコンテンツ タイプを指定するための、Extension Qualifier List (EQL) を作成します。

```
CS150(config)# eql graphics
CS150(config-eql[graphics])#
```

- ステップ 4** EQL の説明として、最長 64 文字の引用符で囲んだテキスト文字列を入力します。

```
CS150(config-eql[graphics])# description "This EQL specifies cacheable graphic files"
```

- ステップ 5** CSS スイッチでキャッシュするコンテンツの拡張子を指定します。1 ~ 8 文字のテキスト文字列を入力します。

```
CS150(config-eql[graphics])# extension jpeg
```

また、拡張子タイプの説明を入力できます。引用符で囲んだテキスト文字列を入力します。最大長は 64 文字です。

```
CS150(config-eql[graphics])# extension jpeg "This is a graphics file"
CS150(config-eql[graphics])# exit
CS150(config)#
```

- ステップ 6** すべてのコンテンツ要求を指定の拡張子と照合するために、コンテンツ ルールに EQL を指定します。

```
CS150(config-owner-content[cisco.com-rule1])# url "/*" eql graphics
```

- ステップ 7** キャッシュ コンテンツ ルールの負荷分散方式を設定します。デフォルトは、ラウンドロビンです。

```
CS150(config-owner-content[cisco.com-rule1])# balance domain
```

- ステップ 8** フェールオーバー タイプ (**bypass**、**linear**、**next**) を指定して、サービスに障害が起きた場合、CSS スイッチがコンテンツ要求を処理する方式を指定します。デフォルトは **linear** です。

```
CS150(config-owner-content[cisco.com-rule1])# failover bypass
```

- ステップ 9** EQL 設定を表示します。

```
CS150(config-owner-content[cisco.com-rule1])# show eql
```

- ステップ 10** キャッシュ設定を示すコンテンツ ルールを表示します。

```
CS150(config-owner-content[cisco.com-rule1])# show rule
```

- ステップ 11** CSS スイッチのコンフィギュレーション モードを終了します。

```
CS150(config-owner-content[cisco.com-rule1])# end
```

■ リダイレクション方式としてのレイヤ 4 スwitチングの設定

ステップ 12 設定を保存します。これで、透過キャッシング サービス用の CSS スwitチの設定は完了です。

```
CS150(config-owner-content[cisco.com-rule1])# copy running-config startup-config
```

ステップ 13 スタンドアロン Content Engine が、CSS スwitチなどのレイヤ 4 対応スwitチからレイヤ 4 によりリダイレクトされたトラフィックを透過的に受信するように設定します。

```
CE100(config)# http 14-switch enable
```

ステップ 14 スタンドアロン Content Engine のコンフィギュレーション モードを終了します。

```
CE100(config)# exit
```

ステップ 15 実行コンフィギュレーションを不揮発性メモリに書き込みます。

```
CE100# write memory
```
