



CHAPTER 6

サーバ ロード バランシングの設定

この章では、Cisco 4700 シリーズ Application Control Engine (ACE) アプライアンス でサーバ ロード バランシングを設定する方法について説明します。この章の構成は、次のとおりです。

- [概要](#)
- [Device Manager GUI を使用したレイヤ7サーバロードバランシングの設定](#)
- [CLI を使用したレイヤ7サーバロードバランシングの設定](#)

概要

この章を読むと、ACE アプライアンスにより提供される基本的なサーバ ロード バランシング機能を理解できます。また、レイヤ7ロードバランシングを目的とする仮想サーバの設定もできるようになります。

クライアントが Web サービスを要求すると、ロード バランシング デバイスにより、要求の送信先サーバが決定されます。たとえば、クライアント要求は、Web ページを求める HTTP GET またはファイルのダウンロードを求める FTP GET から構成することができます。ACE はサーバ ロード バランサとして、クライアント要求に対応できるサーバを、サーバにもサーバファーム全体にも過負荷を与えずに、できるだけ短時間に選択します。

ACE は仮想サーバを使用して、Web サイトへの Web トラフィックを代行受信します。仮想サーバにより、複数の実サーバはロードバランシングを目的とした 1 つのサーバに見えるようになります。仮想サーバはバーチャル IP (VIP) と呼ばれ、IP アドレス、使用されるプロトコル (UDP、TEC など) およびポートアドレスにより定義されます。

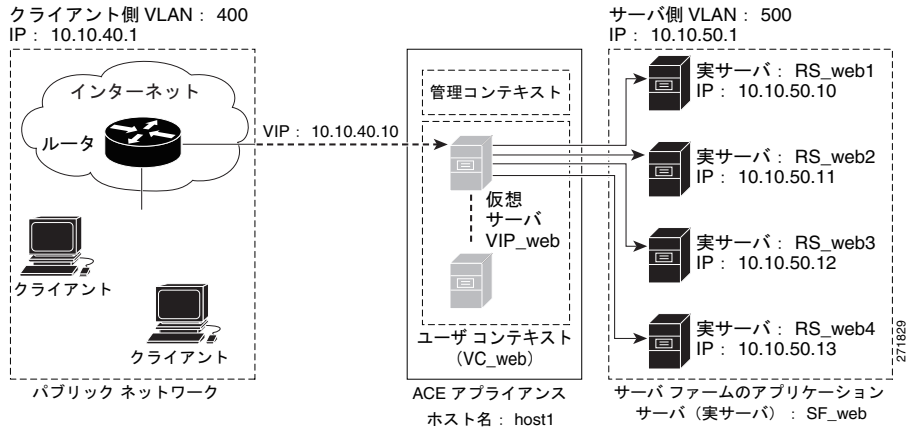
サーバ ファームにグループ化された複数のサーバは、各仮想サーバに割り当てられ、ACE アプライアンスにより、これらの中でロード バランシングが行われます。実サーバは、HTTP や XML コンテンツの配信などのサービスをクライアントに提供するための専用サーバです。サーバ ファームは同じコンテンツを保持し、データ センター内で物理的に同一の場所に設置されます。

次のステップに従って、サーバ ロード バランシングのために ACE を設定します。

- ステップ 1** 仮想サーバを作成します。
- ステップ 2** 実サーバを設定し、このサーバをサーバ ファームと関連付けます。
- ステップ 3** このサーバ ファームを先ほどの仮想サーバに割り当てます。
- ステップ 4** 設定を展開します。

この章では、図 6-1 にあるネットワークの設定例に基づき、Device Manager GUI または Command-Line Interface (CLI; コマンドライン インターフェイス) を使った仮想サーバの設定方法について説明します。

図 6-1 サーバ ロード バランシング の設定例



この例では次のように設定されています。

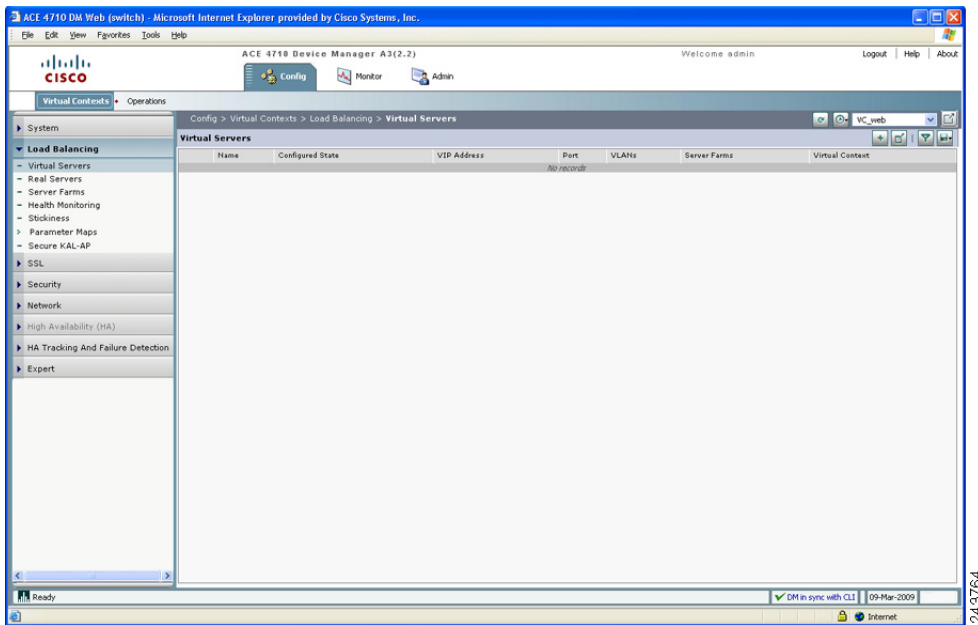
- VLAN 400 からのクライアントトラフィックを VLAN 500 のアプリケーションサーバに送信するために、仮想サーバ VS_web の作成には仮想 IP アドレス 10.10.40.10 が使用されています。
- サーバファーム SF_web には、4 台の実サーバがグループ化されています。
- この仮想サーバはラウンドロビンプレディクタを使用して、サーバファーム内の実サーバの 1 つに向け、クライアント要求を転送します。

Device Manager GUI を使用したレイヤ 7 サーバロードバランシングの設定

次のステップに従って、Device Manager GUI を使用し、レイヤ 7 サーバロードバランシングを設定します。

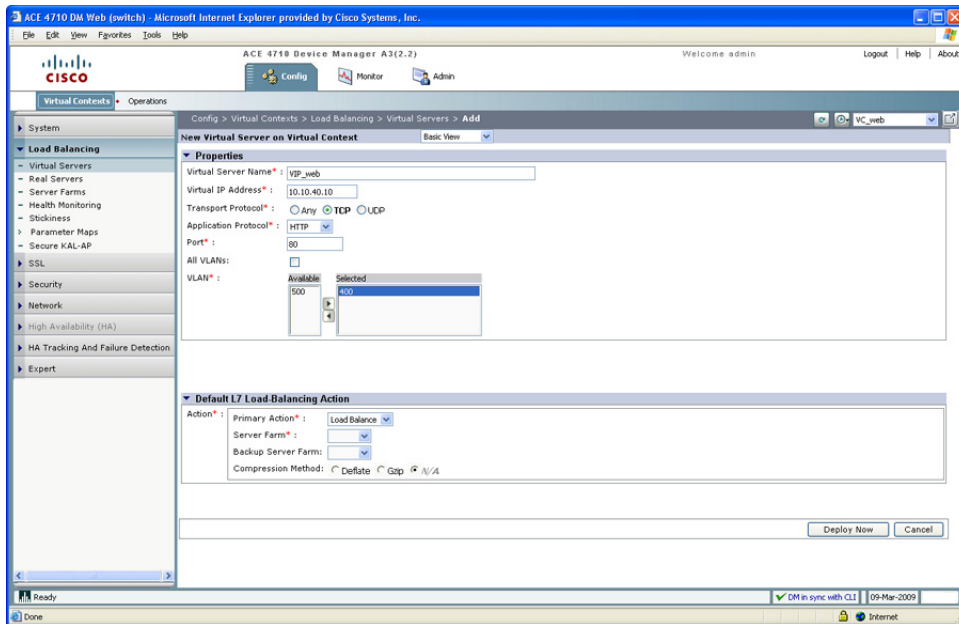
-
- ステップ 1** [Load Balancing] > [Virtual Servers] を選択します。[Virtual Servers] ペインが表示されます (図 6-2)。ユーザ コンテキスト [VC_web] を選択します。

図 6-2 [Virtual Servers] ペイン



ステップ 2 [Add] をクリックして新しい仮想サーバを追加します。[Virtual Server] 設定ウィンドウが表示されます (図 6-3)。

図 6-3 [Virtual Server] 設定ウィンドウのプロパティ



デフォルトでは、[Basic View] 設定オプションが選択され、[Properties] セクションが表示されます。

ステップ 3 [Properties] に次の仮想サーバアトリビュートを入力します。残りのアトリビュートは空白、またはデフォルト値のままにしておきます。

- [Virtual Server Name] : VIP_web
- [Virtual IP Address] : 10.10.40.10



(注) Web サイト (URL) でターゲットされたクライアント要求は、Domain Name System (DNS; ドメイン ネーム システム) に従って IP アドレスに変換されます。仮想サーバに割り当てられた仮想 IP アドレスは、クライアント要求のサービス提供元 Web サイトの URL に対応する IP アドレスです。

- [Transport Protocol] : TCP
- [Application Protocol] : HTTP

■ Device Manager GUI を使用したレイヤ 7 サーバ ロード バランシング の 設定

- [Port] : 80
- [VLAN] : 400

ステップ 4 [Default L7 Load-Balancing Action] セクションの [Primary Action] ドロップダウン リストから [Load Balance] を選択します。

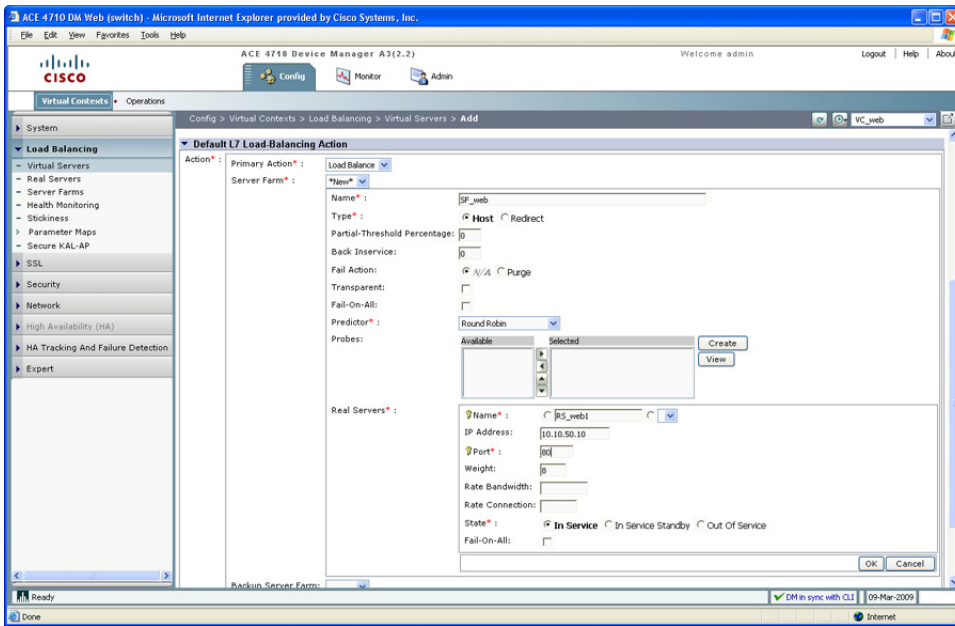
ステップ 5 [Server Farm] ドロップダウン リストから [*New*] を選択し、新規サーバファームを設定します。

ステップ 6 次のサーバファームアトリビュートを入力します。残りのアトリビュートは空白、またはデフォルト値のままにしておきます。

- [Name] : SF_web
- [Type] : Host
- [Predictor] : Round Robin

ステップ 7 [Add] をクリックして、[Real Servers] ペインに新しいエントリを追加します。[Real Servers] ペインに新しいエントリが表示されます (図 6-4)。

図 6-4 [Virtual Server] 設定ウィンドウの [Real Servers] ペイン



24/3751

ステップ 8 設定する 1 つ目の実サーバについて、次のアトリビュートを入力します。残りのアトリビュートは空白、またはデフォルト値のままにしておきます。

- [Name] : RS_web1
- [IP Address] : 10.10.50.10
- [Port] : 80
- [Weight] : 8
- [State] : In Service

[OK] をクリックして、1 つ目の実サーバのアトリビュートを保存します。



(注) ヘルス プローブの設定方法については、[第 10 章「ヘルス プローブを使用したヘルス モニタリングの設定」](#)を参照してください。

ステップ 9 次の実サーバ名および対応する IP アドレスを使用して、ステップ 7 と 8 を繰り返し、[Real Servers] ペインにさらに 3 つのエントリを追加します。残りのアトリビュートはデフォルト値のままにしておきます。

RS_web2 については、次のように入力します。

- [Name] : RS_web2
- [IP Address] : 10.10.50.11
- [Port] : 80

RS_web3 については、次のように入力します。

- [Name] : RS_web3
- [IP Address] : 10.10.50.12
- [Port] : 80

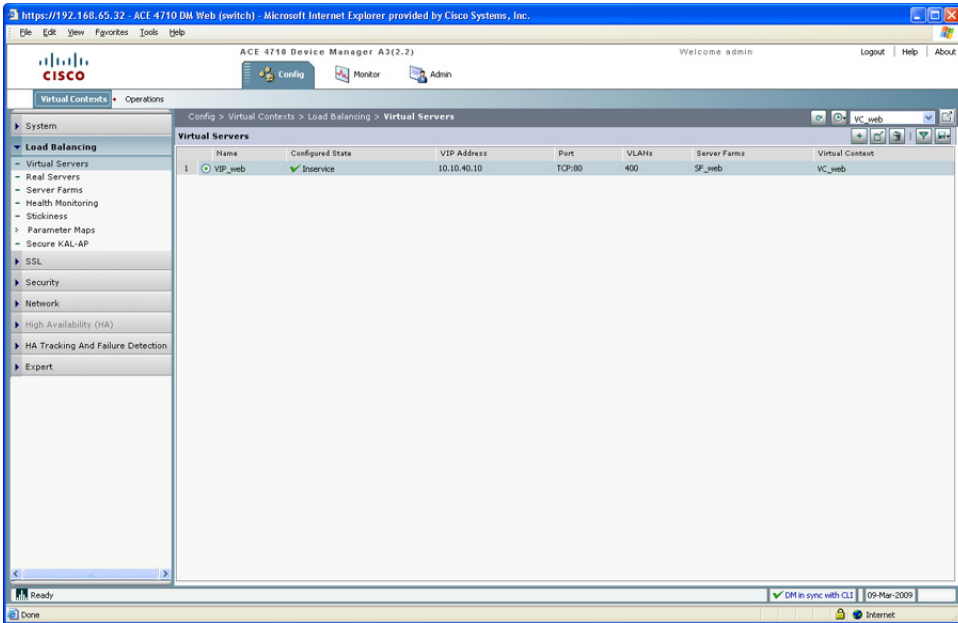
RS_web4 については、次のように入力します。

- [Name] : RS_web4
- [IP Address] : 10.10.50.13
- [Port] : 80

Device Manager GUI を使用したレイヤ 7 サーバ ロード バランシング の設定

- ステップ 10** ウィンドウ下部にある [Deploy Now] をクリックして、仮想サーバに対する設定を保存します。[Virtual Servers] ペインが再度表示されます (図 6-5)。新しく設定された仮想サーバはこのペインに表示されますが、Inservice ステートです。つまり、この仮想サーバは、サーバ ロード バランシング の宛先として使用されています。

図 6-5 作成した仮想サーバが表示されている [Virtual Servers] ペイン



243763

CLI を使用したレイヤ 7 サーバロードバランシングの設定

コマンドライン インターフェイス (CLI) を使用して、レイヤ 7 サーバロードバランシングを設定することができます。ここでは、次の内容について説明します。

- [実サーバの設定](#)
- [サーバファームの作成](#)
- [仮想サーバのトラフィックポリシーの作成](#)

実サーバの設定

次のステップに従って、CLI を使用し、ACE に実サーバを設定します。

- ステップ 1** CLI プロンプトをチェックし、目的のコンテキストで操作が行われていることを確認します。必要に応じて、正しいコンテキストに変更します。

```
host1/Admin# changeto VC_web  
host1/VC_web#
```

- ステップ 2** 設定モードに入ります。

```
host1/VC_web# config
```

- ステップ 3** タイプ ホストとして実サーバ RS_web1 を作成します (デフォルト)。

```
host1/VC_web(config)# rserver RS_web1  
host1/VC_web(config-rserver-host)#
```

- ステップ 4** 実サーバの説明を入力します。

```
host1/VC_web(config-rserver-host)# description content server web-one
```

- ステップ 5** この実サーバに IP アドレス 10.10.50.10 を割り当てます。

```
host1/VC_web(config-rserver-host)# ip address 10.10.50.10
```

- ステップ 6** 実サーバをアクティブにし、設定モードを終了します。

```
host1/VC_web(config-rserver-host)# inervice  
host1/VC_web(config-rserver-host)# exit  
host1/VC_web(config)#
```

ステップ 7 次の実サーバ名、説明、および IP アドレスを使用して、ステップ 3 ~ 6 を繰り返す、さらに 3 台の実サーバを追加します。

RS_web2 については、次のように入力します。

- [Name] : RS_web2
- [Description] : content server web-two
- [IP Address] : 10.10.50.11

RS_web3 については、次のように入力します。

- [Name] : RS_web3
- [Description] : content server web-three
- [IP Address] : 10.10.50.12

RS_web4 については、次のように入力します。

- [Name] : RS_web4
- [Description] : content server web-four
- [IP Address] : 10.10.50.13

ステップ 8 実サーバの設定を表示します。

```
host1/VC_web(config)# do show running-config rserver
```

サーバ ファームの作成

実サーバを作成および構成したら、サーバ ファームを作成し、実サーバと関連付けます。次のステップに従って、サーバ ファームを作成します。

ステップ 1 SF_web という名前でタイプ ホスト (デフォルト) のサーバ ファームを作成します。

```
host1/VC_web(config)# serverfarm SF_web  
host1/VC_web(config-sfarm-host)#
```

ステップ 2 ポート 80 を介して、実サーバ RS_web1 をサーバ ファームに関連付けます。

```
host1/VC_web(config-sfarm-host)# rserver RS_web1 80  
host1/VC_web(config-sfarm-host-rs)#
```

ステップ 3 サーバファーム内で実サーバをアクティブにし、設定モードを終了します。

```
host1/VC_web(config-sfarm-host-rs) # inservice
host1/VC_web(config-sfarm-host-rs) # exit
host1/VC_web(config-sfarm-host) #
```



(注) サーバファーム内の実サーバに接続要求を送信するには、まず、その実サーバをアクティブにする必要があります。そうしないと、ACEはそのサーバが稼働中ではないと見なし、そのサーバファームはクライアントからの要求に対する受信または応答処理を実行できません。

ステップ 4 同様に、実サーバ RS_web2、RS_web3、および RS_web4 を SF_web サーバファームと関連付けます。

```
host1/VC_web(config-sfarm-host) # rserver RS_web2 80
host1/VC_web(config-sfarm-host-rs) # inservice
host1/VC_web(config-sfarm-host-rs) # exit
host1/VC_web(config-sfarm-host) # rserver RS_web3 80
host1/VC_web(config-sfarm-host-rs) # inservice
host1/VC_web(config-sfarm-host-rs) # exit
host1/VC_web(config-sfarm-host) # rserver RS_web4 80
host1/VC_web(config-sfarm-host-rs) # inservice
host1/VC_web(config-sfarm-host-rs) # exit
```

ステップ 5 サーバファーム設定モードを終了します。

```
host1/VC_web(config-sfarm-host) # exit
host1/VC_web(config) #
```

ステップ 6 これらの実サーバの情報を表示し、ネットワーク接続がまだ確立されていないことも、実サーバが動作可能と表示されていることを確認します。

```
host1/VC_web(config) # do show rserver RS_web1
host1/VC_web(config) # do show rserver RS_web2
host1/VC_web(config) # do show rserver RS_web3
host1/VC_web(config) # do show rserver RS_web4
```

ステップ 7 ACE がどのように ARP テーブルに実サーバのデータを入力しているかを表示します。

```
host1/VC_web(config) # do show arp
```

仮想サーバのトラフィック ポリシーの作成

次のステップに従って、ACE で仮想サーバのトラフィック ポリシーを作成します。

- ステップ 1** クラス マップの各エントリを先頭から順番にマッチングしてロード バランシングを行うレイヤ 7 サーバ ロード バランシング ポリシー マップ PM_LB を作成します。

```
host1/VC_web(config)# policy-map type loadbalance first-match PM_LB
host1/VC_web(config-pmap-lb)#
```



(注) ACE は、一連のフロー マッチング基準（トラフィック分類）を指定するために、クラス マップを使用します。ACE は、分類されたインバウンドトラフィックに適用される一連のアクション（機能）を定義するために、ポリシー マップを使用します。

- ステップ 2** 単純なロード バランシング ポリシーでは、ACE にデフォルトのクラス マップを割り当てます。デフォルトのクラス マップには、すべてのトラフィック分類に一致する暗黙の `match any` 文が含まれています。

```
host1/VC_web(config-pmap-lb)# class class-default
host1/VC_web(config-pmap-lb-c)#
```

- ステップ 3** サーバファーム SF_web をレイヤ 7 サーバ ロード バランシング ポリシー マップに追加し、設定モードを終了します。

```
host1/VC_web(config-pmap-lb-c)# serverfarm SF_web
host1/VC_web(config-pmap-c)# exit
host1/VC_web(config-pmap)# exit
host1/VC_web(config)#
```

- ステップ 4** レイヤ 3 および レイヤ 4 ロード バランシング クラス マップ VS_web を作成します。

```
host1/VC_web(config)# class-map VS_web
host1/VC_web(config-cmap)#
```

- ステップ 5** 任意の IP プロトコルの IP アドレス 10.10.40.10 に一致する `match` 文を定義し、設定モードを終了します。

```
host1/VC_web(config-cmap)# match virtual-address 10.10.40.10
255.255.255.0 tcp eq 80
```

```
host1/VC_web(config-cmap) # exit
host1/VC_web(config) #
```

- ステップ 6** レイヤ3とレイヤ4の複数マッチポリシーマップを作成し、分類された着信要求をロードバランシングポリシーマップに振り向けます。

```
host1/VC_web(config) # policy-map multi-match PM_multi_match
host1/VC_web(config-pmap) #
```

- ステップ 7** レイヤ3およびレイヤ4クラスマップVS_webをこのポリシーマップに関連付けます。

```
host1/VC_web(config-pmap) # class VS_web
host1/VC_web(config-pmap-c) #
```

- ステップ 8** レイヤ7ロードバランシングポリシーマップPM_LBをレイヤ3およびレイヤ4ポリシーマップに関連付けます。

```
host1/VC_web(config-pmap-c) # loadbalance policy PM_LB
host1/VC_web(config-pmap-lb-c) #
```

- ステップ 9** ロードバランシング操作についてVIPをイネーブルにし、設定モードを終了します。

```
host1/VC_web(config-pmap-lb-c) # loadbalance vip inservice
host1/VC_web(config-pmap-c) # exit
host1/VC_web(config-pmap) # exit
host1/VC_web(config) #
```

- ステップ 10** 複数マッチポリシーマップの適用先インターフェイスにアクセスします。

```
host1/VC_web(config) # interface vlan 400
host1/VC_web(config-if) #
```

- ステップ 11** 複数マッチポリシーマップPM_multi_matchを適用します。

```
host1/VC_web(config-if) # service-policy input PM_multi_match
host1/VC_web(config-if) # exit
host1/VC_web(config) #
```

- ステップ 12** 実行設定をスタートアップ設定にコピーして保存します。

```
host1/VC_web(config) # do copy running-config startup-config
```

- ステップ 13** PM_multi_matchポリシーマップのサービスポリシー状態を表示します。

```
host1/VC_web(config) # do show service-policy PM_multi_match
```

■ CLI を使用したレイヤ 7 サーバロード バランシングの設定

この章では、ロード バランシング HTTP トラフィック用の仮想サーバを設定しました。次の章では、クライアント要求を適切な実サーバに転送するための、ロード バランシング プレディクタを設定します。