CHAPTER 8

スティッキ性を使用したサーバ持続性の設定

この章では、Cisco 4700 シリーズ Application Control Engine (ACE) アプライアンス でスティッキ性を使ってサーバ持続性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

この章を読むと、ACE アプライアンスがスティッキ性を使用してどのようにサーバ持続性を提供しているか、および HTTP cookie のスティッキ性を設定するにはどうすればいいかを基本的に理解できます。

e- コマース サイトを訪問するお客様は、通常、まず最初にサイトをブラウズします。サイトのアプリケーションによっては、初期接続の確立後にクライアントが 1 つのサーバに固定されることが必要な場合もあれば、ショッピング カートを構築したときなど、クライアントがトランザクションの作成を開始したときに初めて、サーバの固定が必要とされる場合もあります。

たとえば、クライアントがショッピング カートに品物を入れてからは、そのクライアントのすべての要求が同じ実サーバに送信され、すべての品が 1 つのサーバ上の 1 つのショッピング カートに入るようにすることが重要です。お客様のショッピング カートのインスタンスは通常、複数のサーバに重複しているのではなく、特定の Web サーバ上にあります。

同一の実サーバに誘導されるクライアント要求のシーケンスを必要とするタイプのアプリケーションは、e-コマース アプリケーションだけではありません。バンキング アプリケーションやオンライン取引、FTP や HTTP によるファイル転送など、クライアントの情報を維持するような Web アプリケーションはスティッキ性を必要とする可能性があります。

1 つのセッション中、同じクライアントが、複数の同時 TCP または IP 接続、あるいは後続の複数の TCP または IP 接続を同一サーバとの間で維持できるように、ACE を構成できます。ACE におけるこのセッションの持続機能をスティッキ性といいます。セッションとは、クライアントとサーバの間の一定期間（数分から数時間まで）における連続したトランザクションです。

ACE は、設定済みのサーバ ロード バランシング ポリシーに応じて、使用するロード バランシング方式を判断してから、適切なサーバにクライアントを固定します。ACE は、クライアントが特定のサーバにすでに固定されていると判断した場合、ロード バランシング基準に関係なく、そのクライアントの要求をそのサーバに送信します。クライアントが特定のサーバに固定されていないと判断した場合、ACE はその要求に通常のロード バランシング規則を適用します。

特定のクライアントがどのように特定の Web サーバに固定されるか、およびアプリケーションはどのようにクライアントやクライアントのグループを区別しているかを判断するために、ACE では次のスティッキ方式がサポートされています。

- 発信元または宛先 IP アドレス：スティッキ性については、発信元 IP アドレス、宛先 IP アドレス、またはこの両方を使用して、IP ネットワーク マスクに基づいて個々のクライアントとその要求を一意に識別することができます。ただし、企業やサービス プロバイダーがメガプロキシ（無料の匿名 Web プロキシ サーバ）を使用してインターネットへのクライアント接続を確立している場合、送信元 IP アドレスは、要求の真の送信元として信頼できるインジケータにはなりません。このような場合は、セッションの持続性を確実にするためにその他のスティッキ方式を使用します。

- Cookie：クライアントの cookies によって、ACE に接続するクライアント、およびコンテンツ提供サーバに接続するクライアントを一意に識別できます。cookie は、HTTP ヘッダー内の小さなデータ構造です。サーバはこれを使用して、クライアントがその情報を保存するようにという要求とともに、Web クライアントにデータを送信します。この情報には、ユーザがショッピング カートに追加した項目や、選択した旅行日などが含まれることがあります。ACE は、コンテンツ要求を検証し、そのコンテンツがスティッキ状態であると判断すると、そのコンテンツ要求の cookie または URL を調べます。ACE は、cookie または URL 内の情報を使用して、該当するサーバにコンテンツ要求を転送します。
- Hypertext Transfer Protocol（HTTP; ハイパーテキスト転送プロトコル）ヘッダー：HTTP ヘッダーの一意の部分に基づいてスティッキ性を提供するためのヘッダー オフセットを指定することができます。

この e-コマース アプリケーションは、これらのメソッドのどれが特定の e-コマース アプリケーションに適しているかを指示します。

ACE では、スティッキ性アトリビュートのためにスティッキ性グループが使用されます。このようなアトリビュートには、スティッキ方式、タイムアウト、複製、および特定のスティッキ方式に関連するアトリビュートが含まれます。

スティッキ接続を追跡するために、ACE では、スティッキ グループ、スティッキ方式、スティッキ接続、および実サーバに関する情報を持つスティッキ テーブルが使用されます。ACE は設定可能なタイムアウト メカニズムによってスティッキ テーブルのエントリをエージング アウトします。エントリがタイムアウトになると、そのエントリは再利用できる状態になります。接続率が高ければ、スティッキ エントリがタイムアウトになる前にエージング アウトされることもあります。このような場合、ACE は有効期限に最も近いエントリを再利用します。

スティッキ テーブル内のエントリはダイナミック（必要に応じて、ACE が生成）、またはスタティック（設定済み）のいずれかです。スタティック スティッキ エントリを作成すると、ACE はスティッキ テーブル内に即座にそのエントリを置きます。これは設定から削除されるまで、スティッキ データベースにそのまま残ります。

次のステップに従って、スティッキ性を設定します。

ステップ 1　スティッキ性のためにリソースが割り当てられていることを確認します。

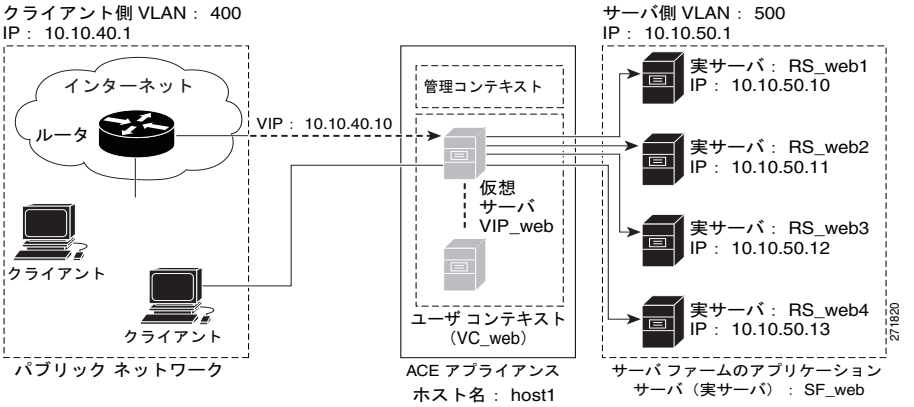
ステップ 2　スティッキ グループを作成します。

ステップ 3　このスティッキ グループを、仮想サーバのレイヤ 7 サーバ ロード バランシング アクションに関連付けます。

ステップ 4　設定を展開します。

図 8-1 は、サーバ ロード バランシング環境で、クライアントからの要求が、あるセッションの実サーバ RS_web4 に固定されていることを図に表したものです。

図 8-1　サーバに固定されているクライアント要求



この章では、HTTP cookie スティッキ方式を使用してスティッキ性を設定する方法について説明します。IP アドレスと HTTP ヘッダー方式を使用してスティッキ性を設定する方法については、『*Cisco 4700 Series Application Control Engine Appliance Server Load-Balancing Configuration Guide*』を参照してください。

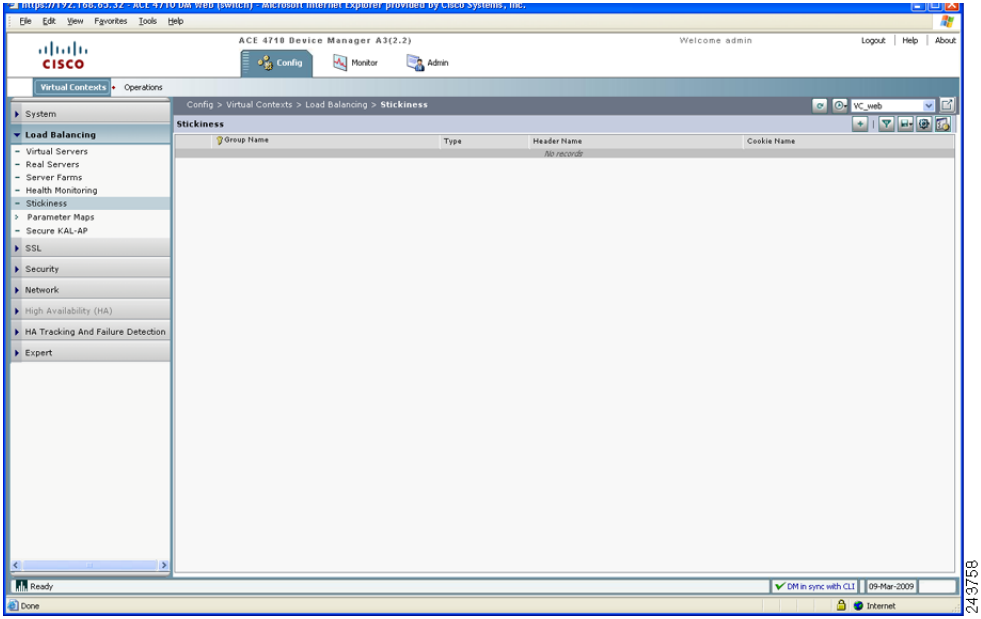
Device Manager GUI を使用した HTTP cookie のスティッキ性の設定

次のステップに従って、GUI を使用し、この HTTP cookie のスティッキ性を設定します。

ステップ 1　スティッキ グループを設定するコンテキストが、リソースをスティッキ機能に割り当てるようなリソース クラスに関連付けられていることを確認します。第 3 章の 「リソース クラスの作成」セクションを参照してください。

ステップ 2　[Load Balancing] > [Stickiness] を選択します。[Stickiness] ペインが表示されます（図 8-2）。

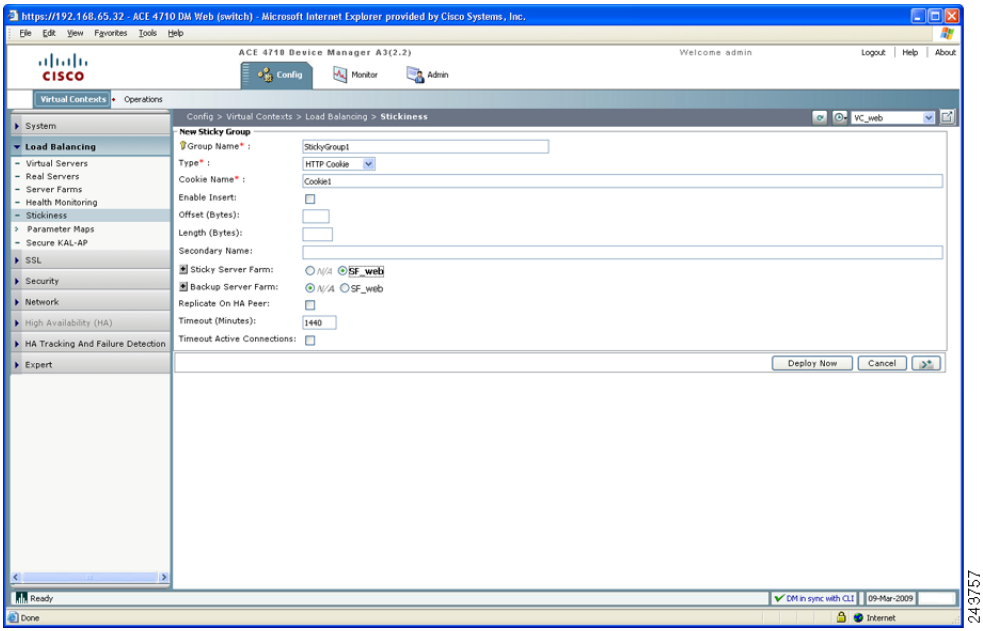
図 8-2　[Stickiness] ペイン



ステップ 3　[VC_web] コンテキストを選択します。

ステップ 4　[Add] をクリックして、新しいスティッキ グループを追加します。[Stickiness] 設定ウィンドウが表示されます（図 8-3）。

図 8-3　[Stickiness] 設定ウィンドウ



ステップ 5　新しいスティッキ グループについて、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。

- [Group Name]：StickyGroup1
- [Type]：HTTP Cookie
- [Cookie Name]：Cookie1
- [Sticky Server Farm]：SF_web

ステップ 6　[Deploy Now] をクリックして、[Stickiness] ペインに新しいスティッキ グループを追加します。

CLI を使用した HTTP cookie のスティッキ性の設定

次のステップに従って、Command-Line Interface（CLI; コマンドライン インターフェイス）を使用し、この HTTP cookie のスティッキ性を設定します。

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
host1/VC_web(config)#
```

ステップ 3　HTTP cookie タイプのスティッキ グループを作成し、cookie 設定モードに入ります。

```
host1/VC_web(config)# sticky http-cookie Cookie1 StickyGroup1
host1/VC_web(config-sticky-cookie)#
```

ステップ 4　HTTP cookie スティッキ性に対するタイムアウトを設定します。

```
host1/VC_web(config-sticky-cookie)# timeout 1440
```

ステップ 5　このスティッキ グループにサーバ ファームを関連付け、設定モードを終了します。

```
host1/VC_web(config-sticky-cookie)# serverfarm SF_web
host1/VC_web(config-sticky-cookie)# exit
host1/VC_web(config)# exit
host1/VC_web#
```

ステップ 6　HTTP cookie 設定を表示します。

```
host1/VC_web# show running-config sticky
```

この章では、HTTP-cookie 方式を使用して、スティッキ グループを設定しました。次の章では、SSL セキュリティを設定します。