



WebOTX SIP Application Server

BIG-IP Local Traffic Manager

連携システム構築ガイド

2007.12.20

1 版

NEC

第二システムソフトウェア事業部

改版履歴

版数	年月日	改訂内容	備考
ドラフト 1.0	2007/10/2	ドラフト 1 版として発行	
ドラフト 2.0	2007/10/09	全体 文章の記述ミスを修正	
		2. 連携方式 冒頭文章を修正	
		3. 1. IP アドレスの設定、3. 2. Pool 作成、3. 3. 仮想サーバの作成 冒頭文章に目的を追記	
		4. SIP 特有の設定 冒頭文章を修正	
		4. 1. 仮想サーバの設定、4. 3. iRule 作成 冒頭文章に目的を追加	
		5. SIP シーケンス詳細 冒頭文章を追加	
		6. おわりに 章を追加	
ドラフト 3.0	2007/11/09	4. 1. 仮想サーバの設定 元を”2.”に移動し、“1.”を追加	
		図 4-2 を変更	
		以下の図中における BIG-IP 画像を修正 図 2-1、図 2-2、図 2-3、図 3-1、図 5-1、図 5-2、図 5-3、図 5-4	
1 版	2007/12/20	1 版として発行	

目次

1. はじめに	2
1.1. 本書の目的	2
2. 連携方式	3
2.1. 連携の目的	3
2.2. 連携構成	4
2.3. セッション情報の管理	5
3. 環境設定	6
3.1. IPアドレスの設定	6
3.2. Pool作成	10
3.3. 仮想サーバの作成	12
4. SIP特有の設定	14
4.1. 仮想サーバの設定	14
4.2. SIPプロファイルの登録	15
4.3. iRule作成	17
5. SIPシーケンス詳細	20
5.1. 登録処理	20
5.2. ローカルネットワーク発信-外部ネットワーク着信処理	22
5.3. 外部ネットワーク発信-ローカルネットワーク着信処理	24
5.4. ローカルネットワーク発信-ローカルネットワーク着信SIPシーケンス	26
6. おわりに	28

1. はじめに

1.1. 本書の目的

本書はBIG-IP Local Traffic Manager と Web0TX SIP Application Server 間の連携システムを構築するためのガイドとして記述します。

本書に記載のBIG-IP Local Traffic Managerはバージョン9.4.2¹を、Web0TX SIP Application ServerはV7.1を使用しています。

¹ 今後のバージョンアップにより、iRules や SIP パーシステントの設定内容が変わる可能性があります。

2. 連携方式

BIG-IP Local Traffic Manager と Web0TX SIP Application Server を連携する目的、および連携を実現するための構成を本章で記載します。

2.1. 連携の目的

BIG-IP Local Traffic Manager と Web0TX SIP Application Server を連携することにより、SIP の負荷分散が可能となります。

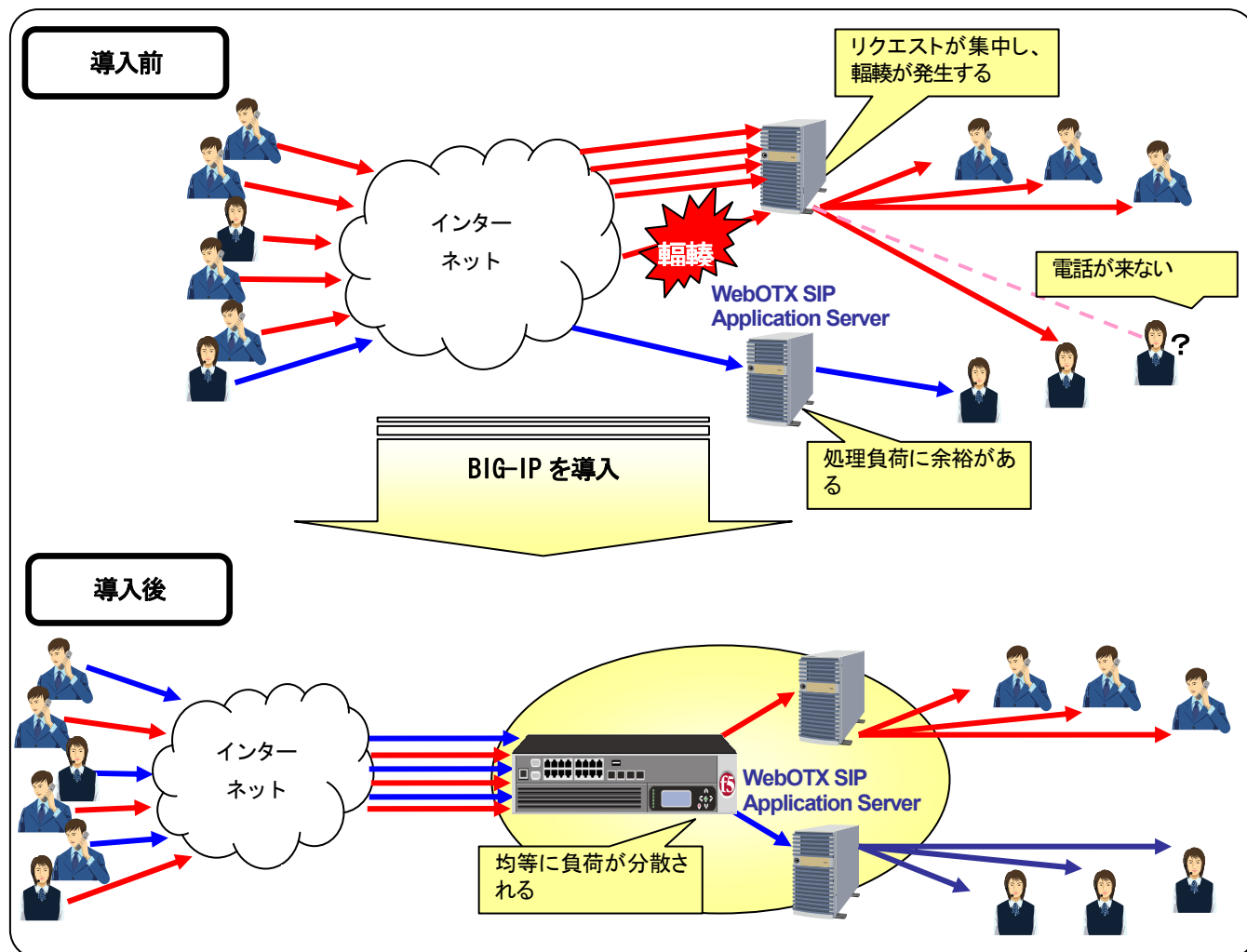


図 2-1 BIG-IP/Web0TX 連携の目的

複数の Web0TX SIP Application Server で構成されたシステムでは、他の Web0TX SIP Application Server の処理能力には余裕があっても、特定の Web0TX SIP Application Server にリクエストが集中し輻輳が発生することがあります。

BIG-IP Local Traffic Manager との連携による負荷分散機能の導入により、複数の Web0TX SIP Application Server のリソースをシステム全体でより効率的に運用することが可能になります。

2. 2. 連携構成

BIG-IP Local Traffic ManagerとWeb0TX SIP Application Serverを同一ネットワーク内に配置した連携構成図を図 2-2に示します。

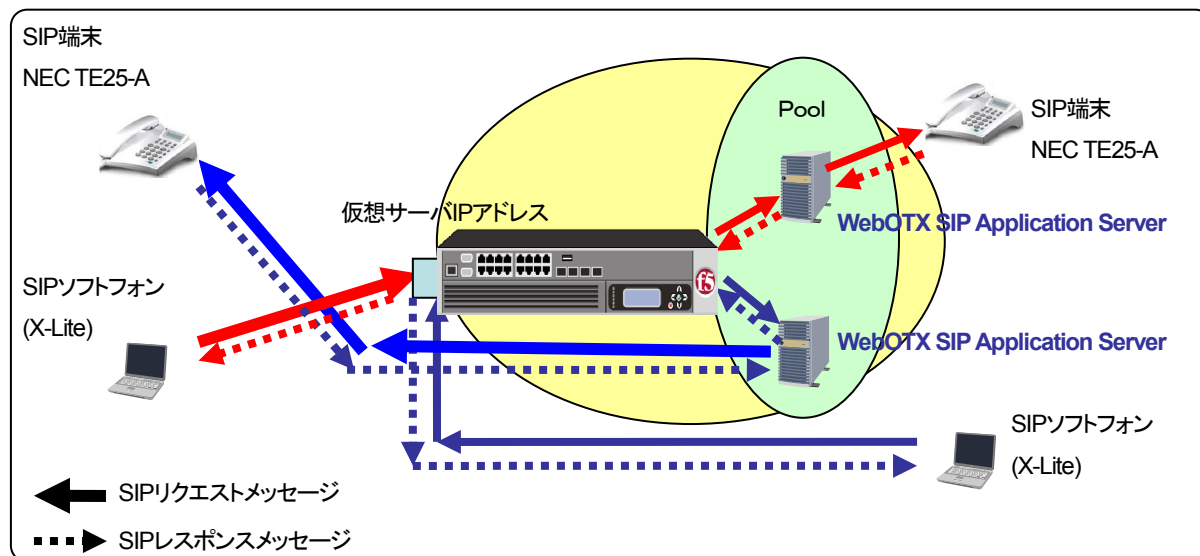


図 2-2 BIG-IP/Web0TX 連携構成図

BIG-IP仮想サーバの同一Poolに、UAとして動作するWeb0TX SIP Application Serverとプロキシサーバとして動作するWeb0TX SIP Application Serverを混在させることはできません。必ず、どちらかに統一する必要があります。図 2-2は、Web0TX SIP Application Serverをプロキシサーバとして使用した場合の構成図です。

SIP リクエストは、ローカルネットワークから送られたか外部ネットワークから送られたかに関わらず、常に BIG-IP の仮想サーバ IP アドレスを経由し、BIG-IP 管理用 IP アドレスから Web0TX SIP Application Server に受信されるように構成します。このため、SIP リクエストがBIG-IP Local Traffic Manager 以外から Web0TX SIP Application Server に送信されることはありません。

2.3. セッション情報の管理

同一セッション内のリクエストを、常に決まった Pool メンバに振り分けるために、BIG-IP Local Traffic Manager は、INVITE リクエスト受信時（セッション確立時）に、Call-ID 毎に振り分け先を保持します（本機能を SIP パーシステンスと呼びます）。

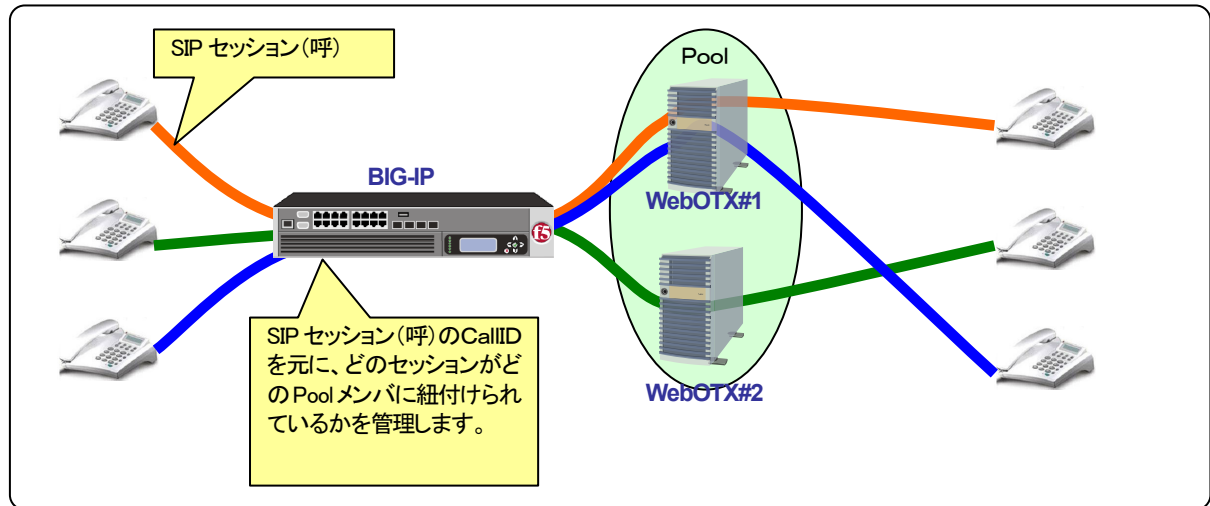


図 2-3 セッション情報の管理

HTTP セッションの場合、Web サーバと Web クライアントの関係は、常に同じであり、入れ替わることがありません。このため、セッションの確立要求を行ったホスト (HTTP では Web クライアント) のアドレスで振り分け先を管理する「Src IP アドレスパーシステンス」を使用することができます。

これに対し、SIP セッションの場合、リクエストごとにサーバとクライアントの役割が入れ替わる可能性があるため、「Src IP アドレスパーシステンス」は使用せず、代わりに Call-ID により振り分け先を管理する「SIP パーシステンス」を使用します。

3. 環境設定

本項ではBIG-IPの環境設定方法について記載します。図 3-1に各設定項目と、それを記述する章を示します。

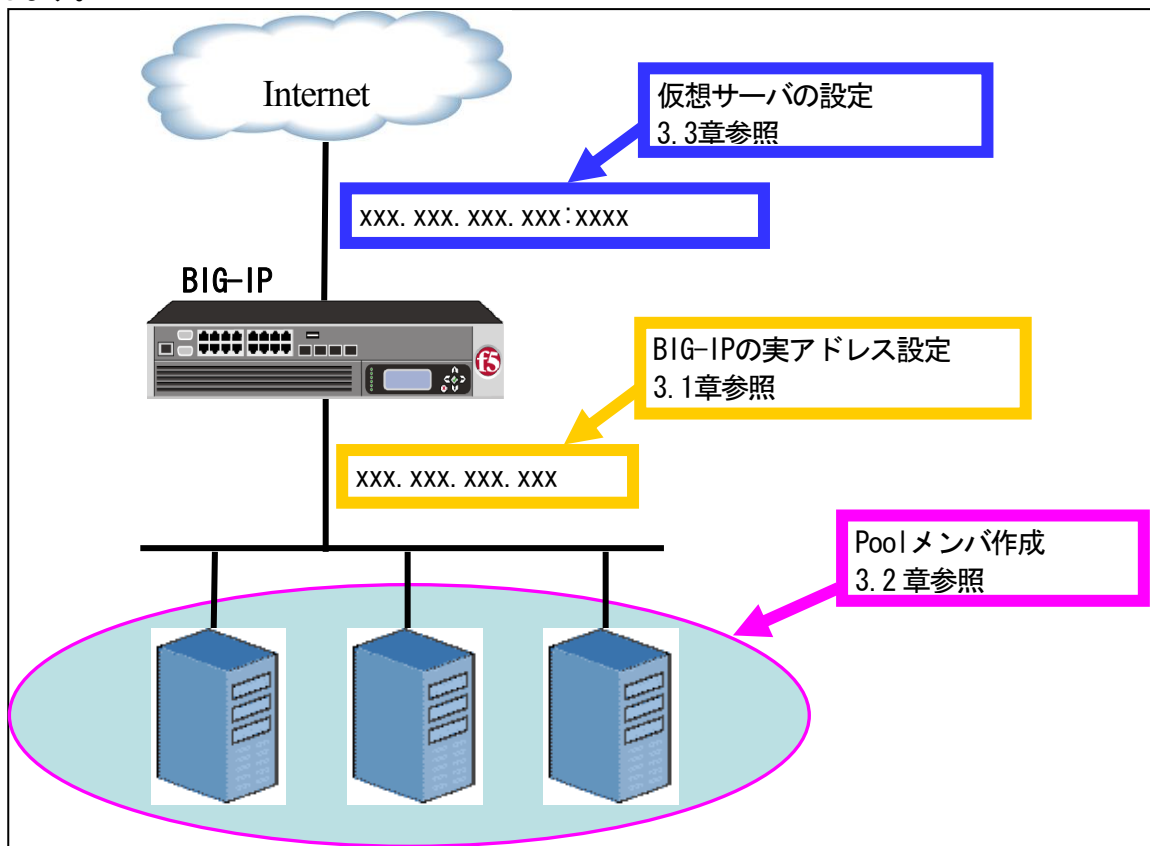


図 3-1 環境設定

3.1. IPアドレスの設定

本項では、BIG-IP Local Traffic Manager で使用する IP アドレスの設定を行います。ここで設定した IP アドレスを使用することで、同一ネットワーク上の端末から BIG-IP Local Traffic Manager へアクセスすることが可能になります。

1. 管理用の LAN ポートを使用し、BIG-IP Local Traffic Manager にアクセスします。”Network”から”VLANs”を選択し、”create”ボタンを押下します。

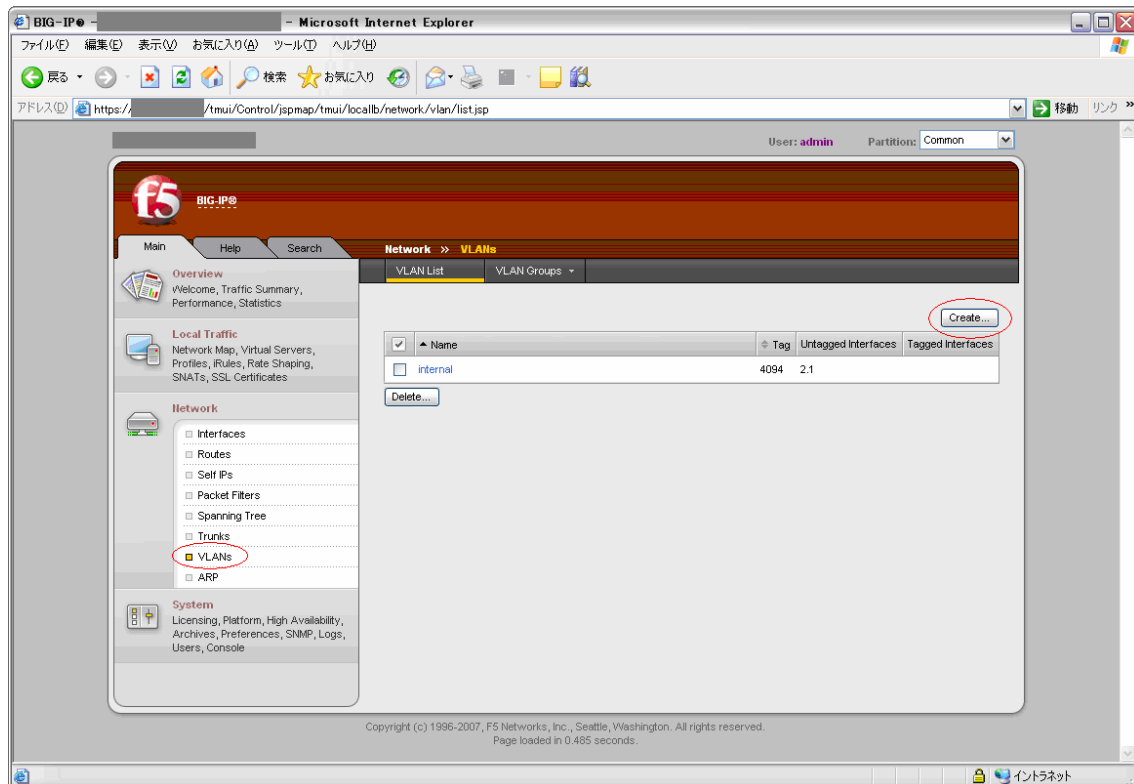


図 3-2 IP アドレスの設定 1

2. "Name"欄に、VLAN の名前を入力します。

"Interfaces"欄の" Available"から、使用する LAN ポートの番号を選択し、"Untagged"へ移動させ、"Finished"ボタンを押下します。画面遷移後に、追加した VLAN の名前が表示されれば、正常に作成されています。

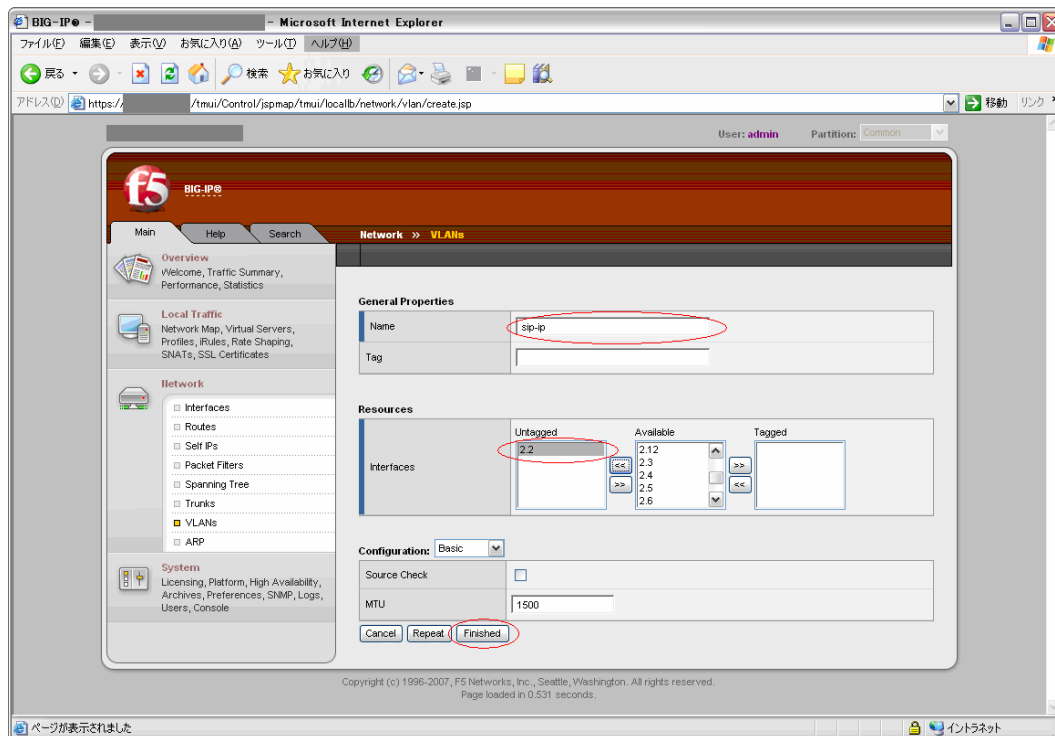


図 3-3 IP アドレスの設定 2

3. "Network"から"Self IPs"を選択し"create"ボタンを押下します。

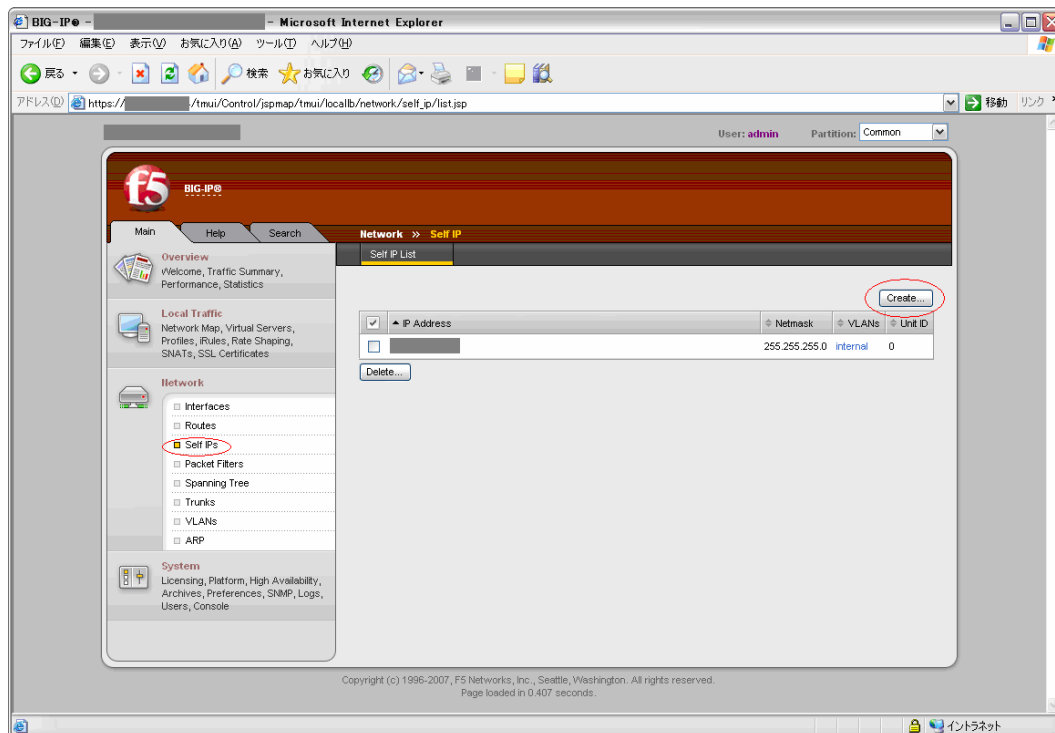


図 3-4 IP アドレスの設定 3

4. "VLAN"に 2 で設定した名前を選択します。"IP Address"と"Netmask"に IP アドレスとサブネットマスクを入力し、"Finished"ボタンを押下します。

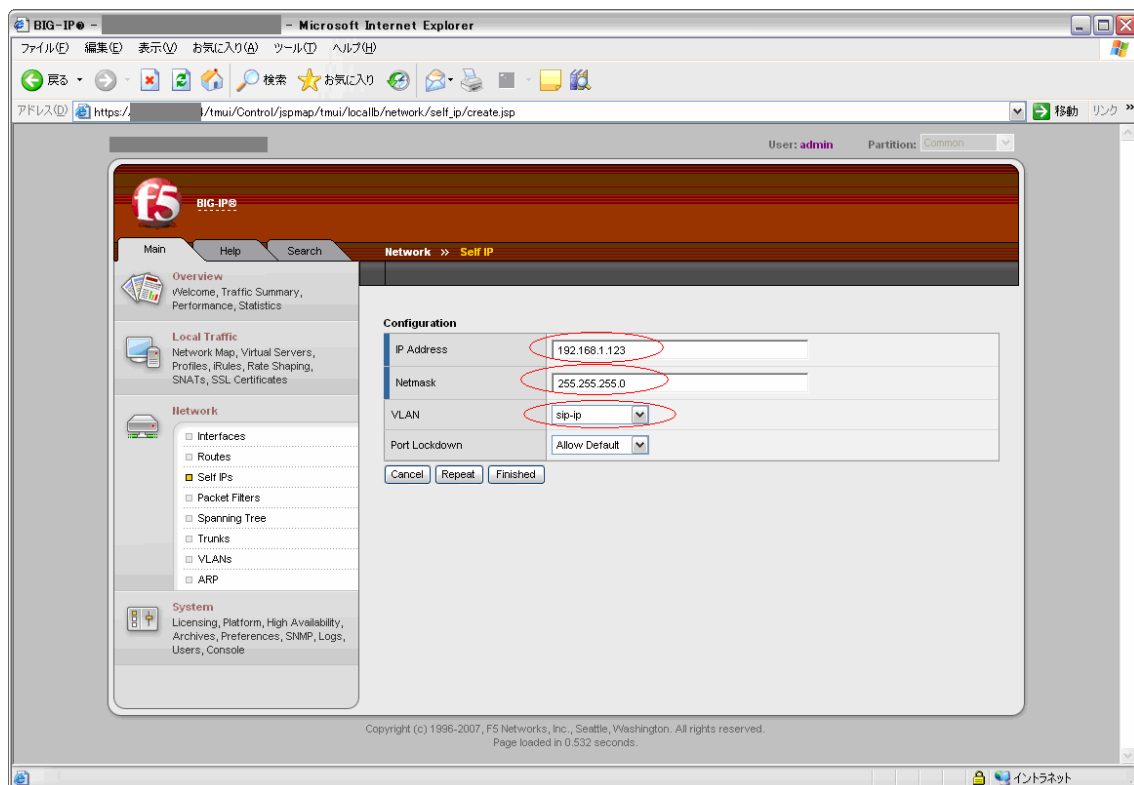


図 3-5 IP アドレスの設定 4

5. 2 で設定した LAN ポートにケーブルを繋ぎ、4 で設定した IP アドレスに向かって、同一ネットワーク上の端末から、BIG-IP Local Traffic Manager の管理コンソールへアクセス出来れば、設定完了です（画面の設定例の様に IP アドレスを”192. 168. 1. 123”にした場合、<https://192. 168. 1. 123> にアクセスできれば設定完了）。

3.2. Pool作成

本項では、BIG-IP Local Traffic Manager が負荷分散を実施するホストを設定します。

管理用のLANポート、または3.1章で設定したIPアドレスからBIG-IP Local Traffic Managerの管理コンソールへアクセスし、設定を行います。

1. "Local Traffic"から"Pools"を選択し、"create"ボタンを押下します。

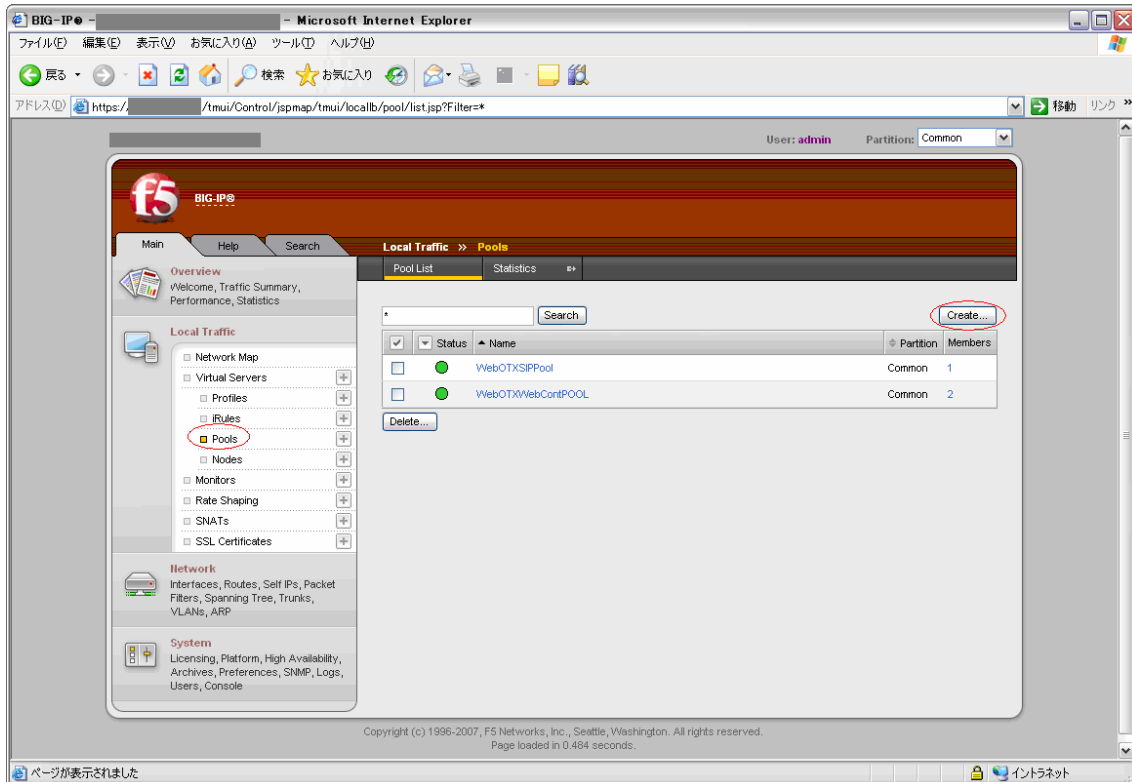


図 3-6 Pool 作成 1

2. "Name"欄に Pool 名を記入し、"Health Monitors"欄に作成するプールで使用したいモニタを"Active"に移動します。

"Load Balancing Method"欄にロードバランスのモードを選択します。

"New Members"欄に Pool メンバとして登録するホストの IP アドレスとポート番号を入力し、"Add"ボタンを押下します(下記設定例では、"192.168.1.2:5060"と"192.168.1.3:5061"を登録している)。Finished ボタンを押下し、Pool 設定を完了します。

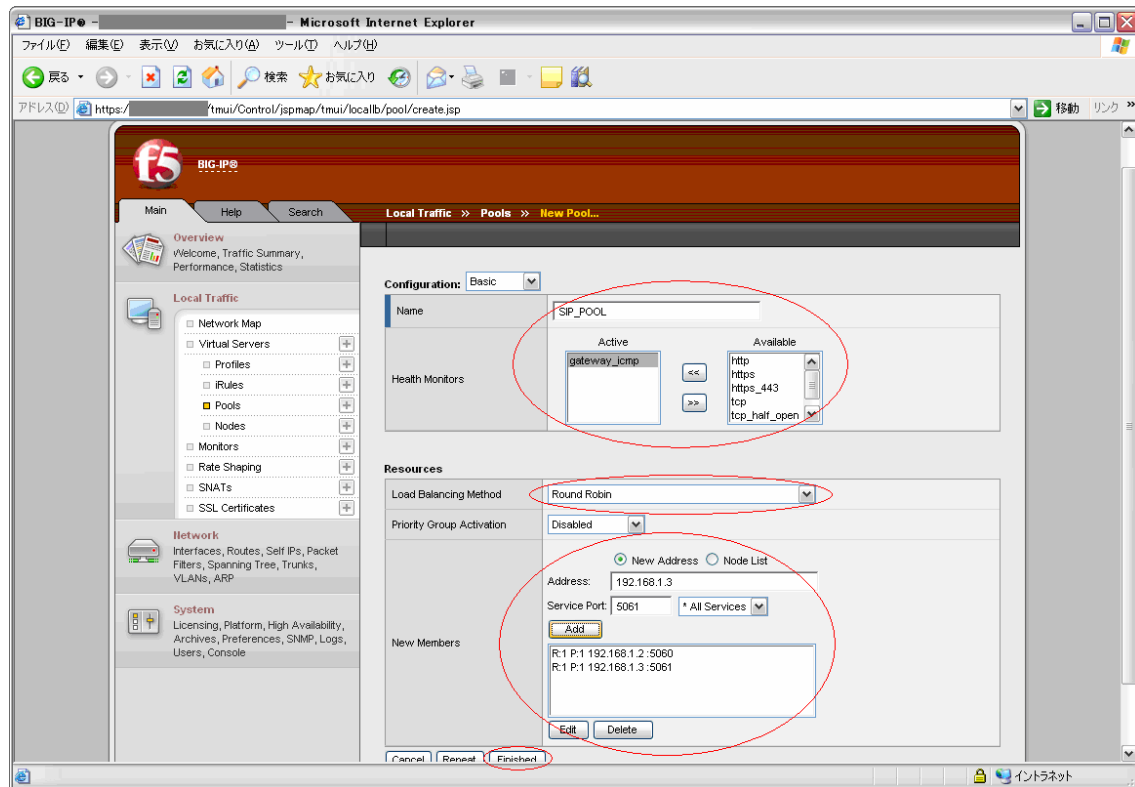


図 3-7 Pool 作成2

3.3. 仮想サーバの作成

本項は、外部ネットワークに公開するBIG-IP Local Traffic ManagerのIPアドレスを設定します。また、設定したIPアドレスに対して要求があった場合に、「3.2. Pool作成」で作成したPoolメンバに振り分けられるように設定します。

「3.2. Pool作成」と同様の手段を使用してBIG-IP Local Traffic Managerの管理コンソールから設定を行います。

1. "Local Traffic"から"Virtual Servers"を選択し、"create"ボタンを押下します。

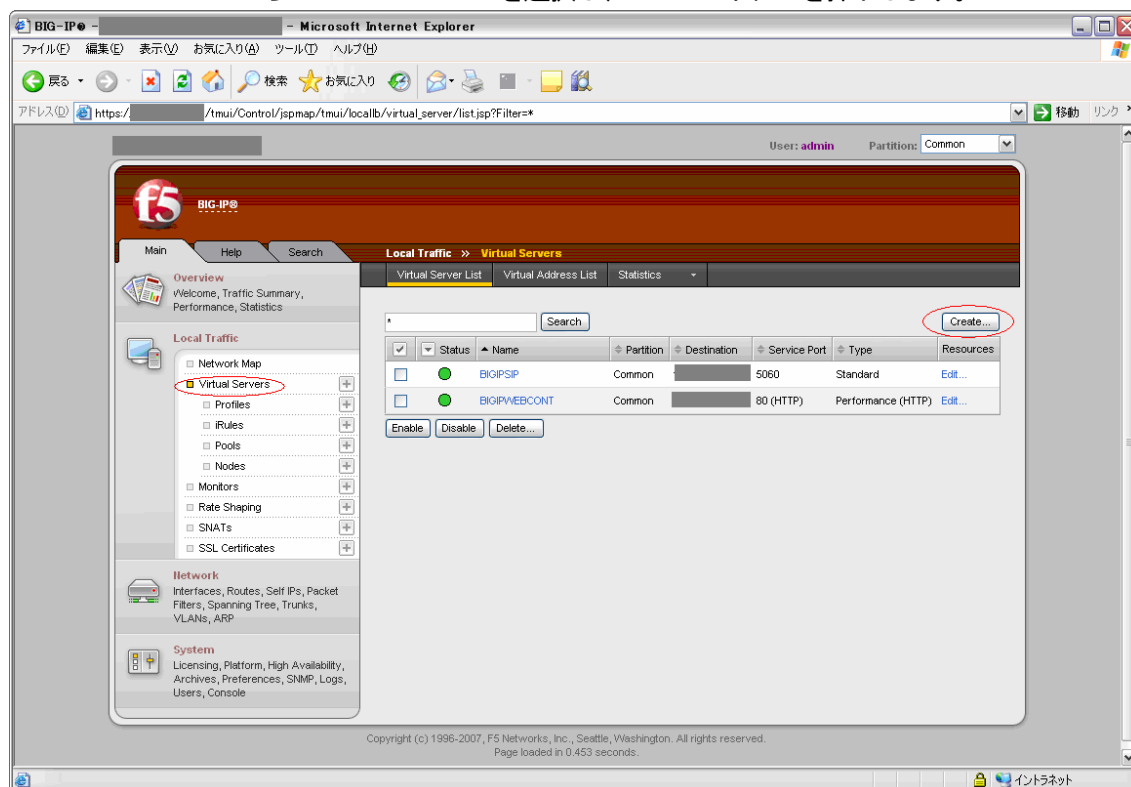


図 3-8 仮想サーバ作成 1

2. "Name"欄に仮想サーバの名前を入力します。"Destination"欄に、仮想サーバに割り当てる IP アドレスとポート番号を設定します。作成する仮想サーバを無効にする場合は"State"を"Disabled"に設定します。

仮想サーバの作成目的に合わせ、"Configuration"項目の各種設定を行います。

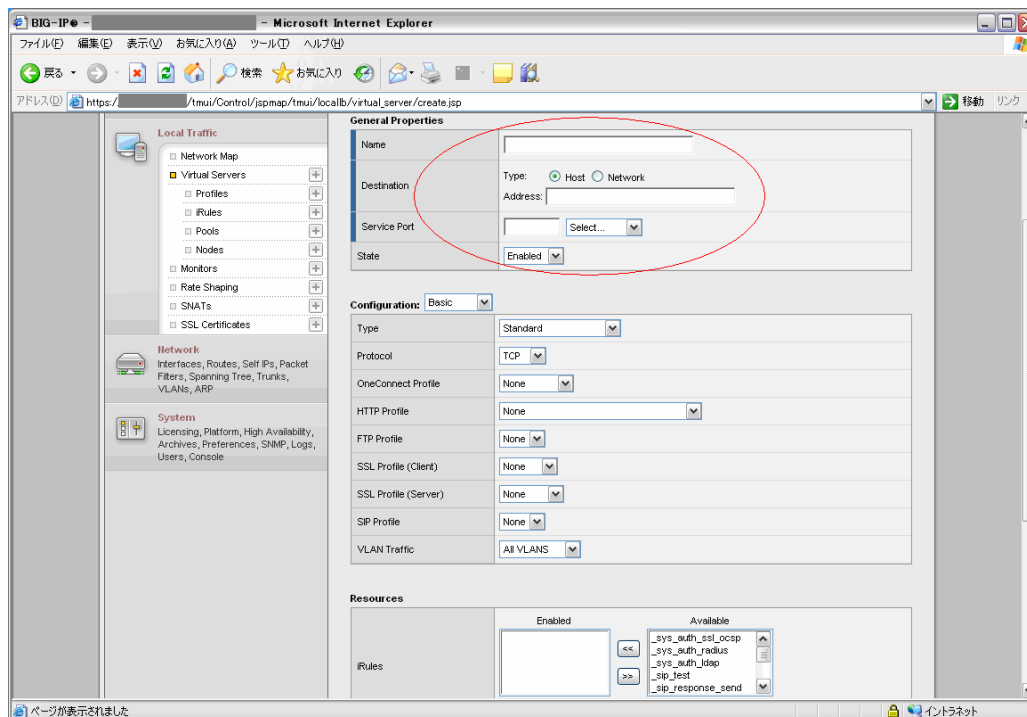


図 3-9 仮想サーバ作成2

- 仮想サーバにiRuleが必要であれば、「iRules」欄の「Enabled」に「Available」中のiRuleを移動します。「Default Pool」欄に、「Pool」で作成したPool名を選択し、「Finished」ボタンを押下します。（必要になった際に追加変更も可能なため、ここでは何も選択しなくても構わない）

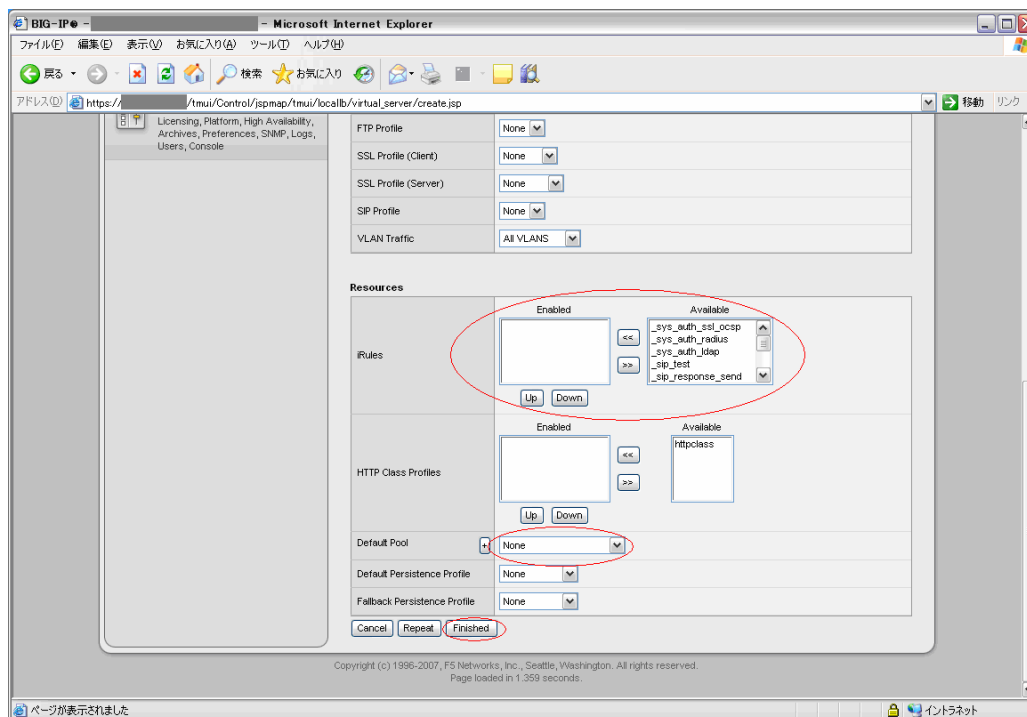


図 3-10 仮想サーバ作成3

4. SIP特有の設定

BIG-IP Local Traffic Manager がSIP メッセージを負荷分散するために必要になる SIP 特有の設定について、以下に記載します。

4.1. 仮想サーバの設定

1. “Virtual Servers”の設定画面から、“Protocol”を”UDP”に変更します。

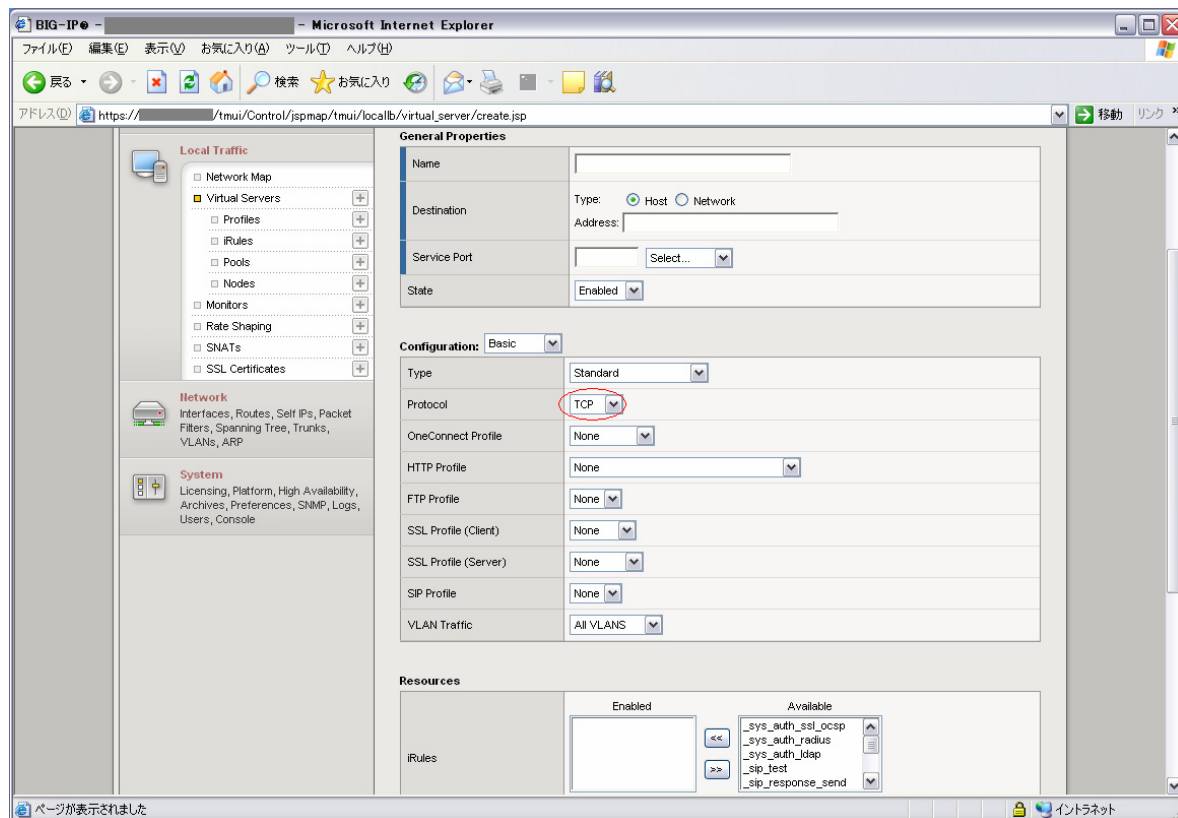


図 4-1 SIP 仮想サーバの設定 1

2. “Default Persistence Profile”を”SIP info”に変更し、“Finished”ボタンを押下します。これにより、SIP リクエスト受信時に、SIP パーシステンスが動作します。

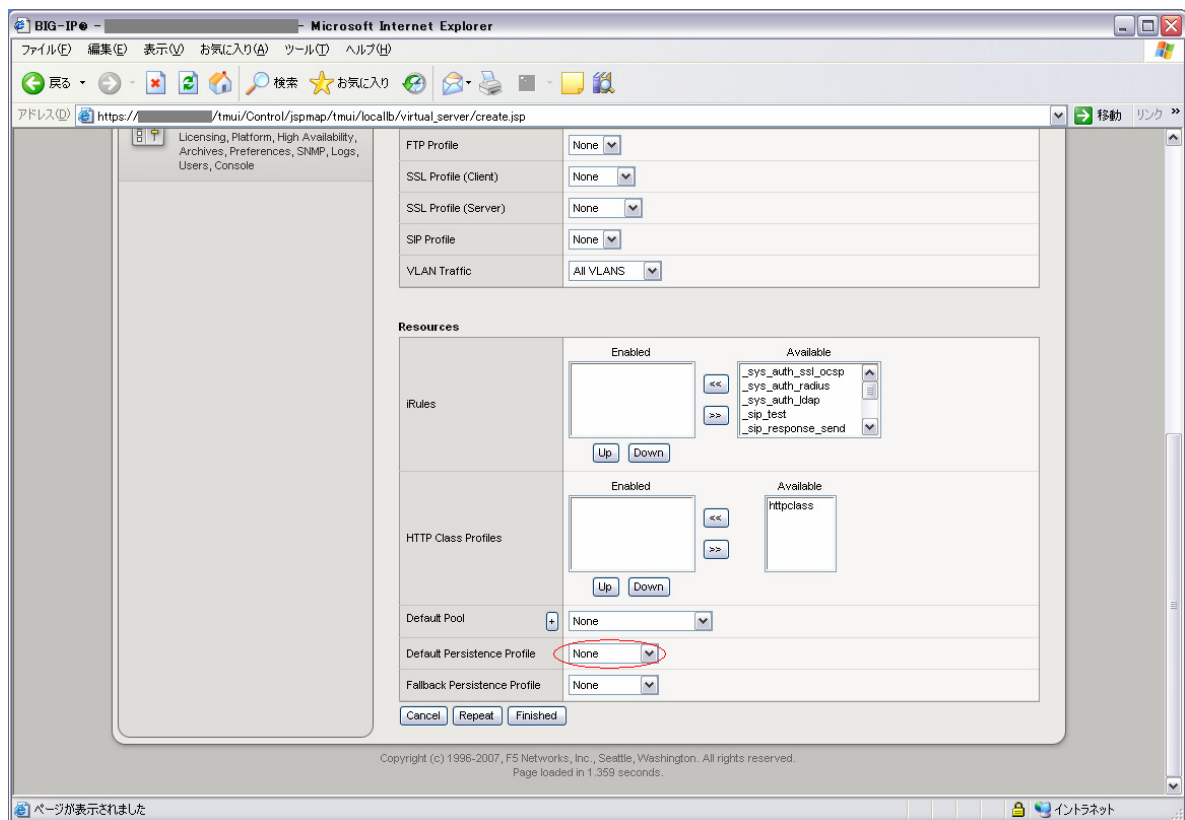


図 4-2 SIP 仮想サーバの設定 2

4.2. SIPプロファイルの登録

1. "Local Traffic"から"Profiles"を選択し、"Services"の"SIP"を選択します。

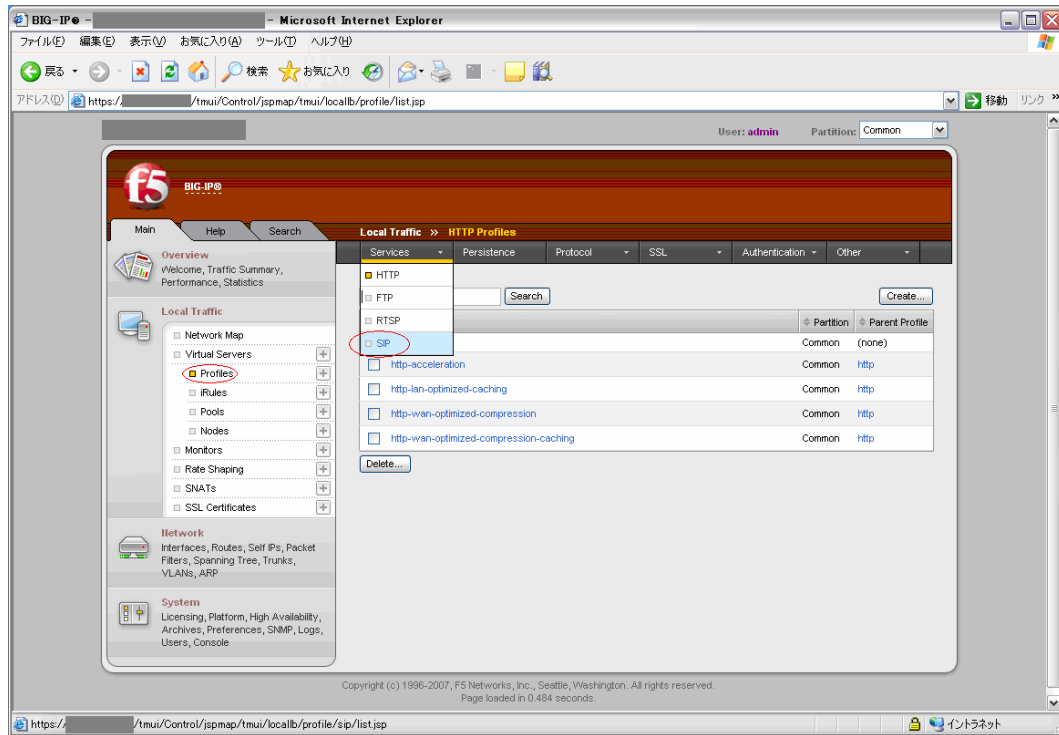


図 4-3 SIP プロファイルの登録 1

2. 画面上の”sip”を選択します。

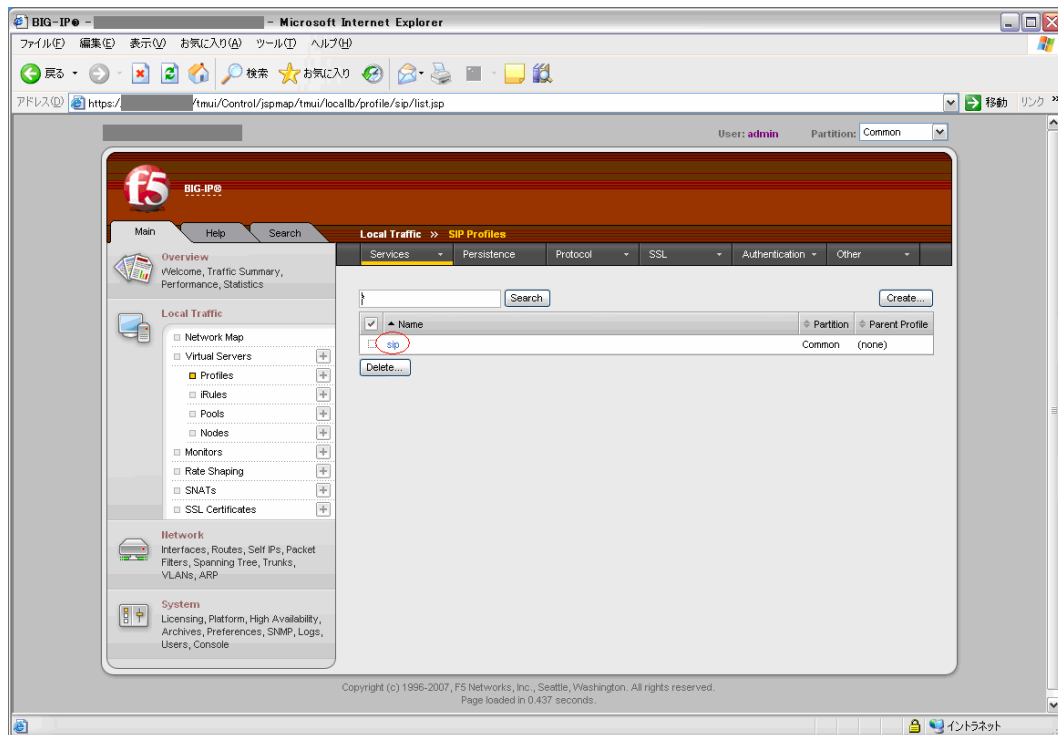


図 4-4 SIP プロファイルの登録2

3. ”Settings”の”Terminate on BYE”欄のチェックボックスを外します。

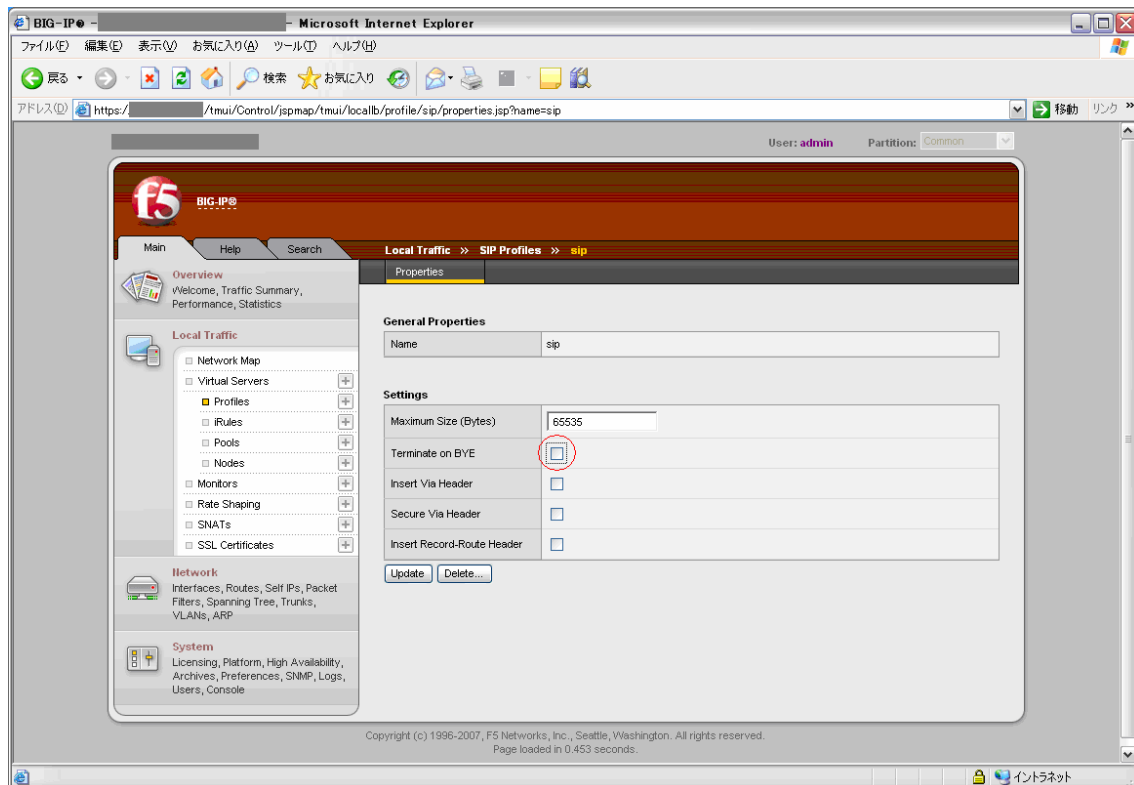


図 4-5 SIP プロファイルの登録3

4.3. iRule作成

“Local Traffic”の”iRules”を選択し、iRule を記述します。以下に SIP iRule を記載します。下記2つの iRule を SIP 用仮想サーバに設定します。iRule は、BIG-IP Local Traffic Manager が SIP リクエストを正しく WebOTX SIP Application Server にプロキシするために必要となります。

尚、本 iRule はサンプルのため、要件に応じて別途変更を加える必要があります。

• _sip_request_send_rule

```

when SIP_REQUEST_SEND {
    log local0.info "in"
    # Via ヘッダを追加。
    SIP::header insert Via [format "SIP/2.0/UDP %s:%s:branch=%s" [IP::local_addr] [UDP::local_port]
[SIP::via branch]]

    # リクエストにRoute ヘッダが含まれていなければ、イニシャルリクエスト
    # と判断する。
    # サブシーケントリクエストであれば、先頭のRoute ヘッダを削除する
    if { [SIP::header Route] = "" }
    {
        set record_route "<sip: xxx. xxx. xxx. xxx:5060;lr>"
        SIP::header insert Record-Route $record_route
    }
    else
    {
        if { [SIP::header Route 1] != "" }
        {
            SIP::header remove Route 0
        }
        else
        {
            set route_value [SIP::header Route]
            if { [string first "," $route_value] = -1 }
            {
                SIP::header remove Route 0
                SIP::header insert Route [format "<sip:%s:5060;lr>" [IP::server_addr]]
            }
            else
            {
                set new_route [string range $route_value [string first "," $route_value] 65535]
                set new_route_2 [format "<sip:%s:5060;lr>%s" [IP::server_addr] $new_route]
                SIP::header remove Route 0
                SIP::header insert Route $new_route_2
            }
        }
    }
}

```

仮想サーバの IP アドレス
を記述する。

• _sip_response_send_rule

```
when SIP_RESPONSE {
  log local0.info "remove header"
  log local0.info "response code = [SIP::response code]"

  # Via ヘッダ除去作業
  # ・Via ヘッダが複数個あれば、先頭のVia ヘッダを削除する。
  # ・Via ヘッダが1 個しかない場合、最初に登場するパラメータを削除する。
  if { [SIP::header Via 1] != "" }
  {
    SIP::header remove Via 0
  }
  else
  {
    set via_value [SIP::via]
    set new_via [string range $via_value [string first "," $via_value] 65535]
    set new_via_2 [string range $new_via [string first "S" $new_via] 65535]
    SIP::header remove Via 0
    SIP::header insert Via $new_via_2
  }
  log local0.info "Via = [SIP::via]"
}
```

5. SIPシーケンス詳細

図 2-2の連携構成時に流れるSIPシーケンスを本章に記載します。また、SIPコマンド受信時に行う処理を説明します。

5.1. 登録処理

登録処理時の SIP シーケンスを図 5-1 に示します。

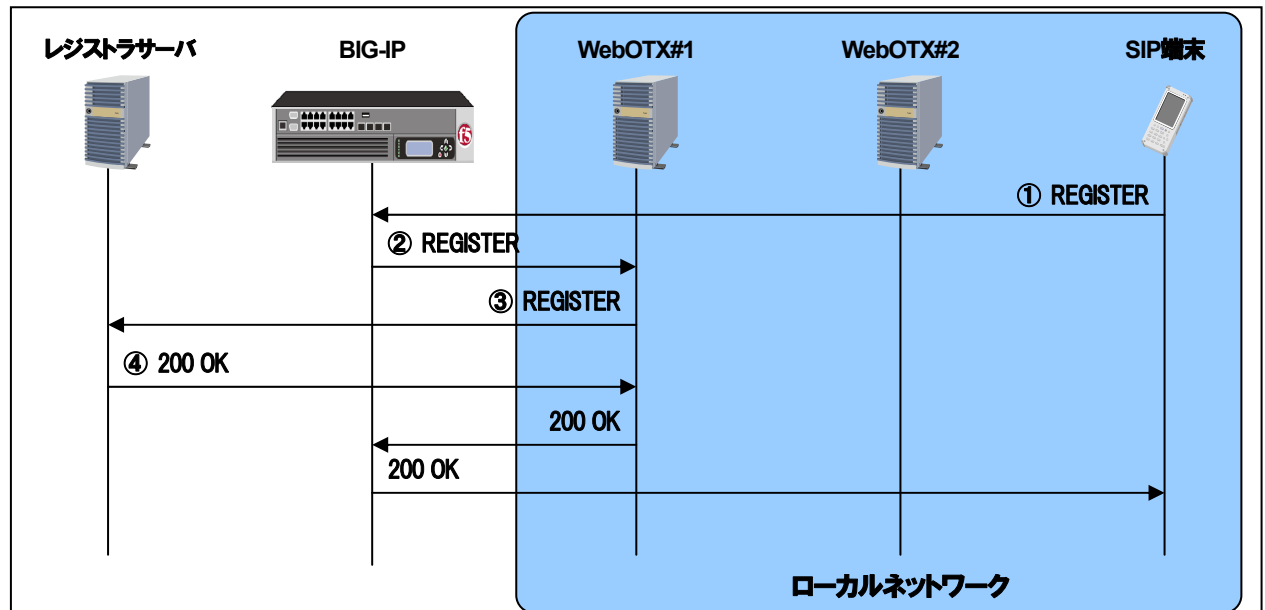


図 5-1 登録処理シーケンス

- ①端末はREGISTER リクエストをBIG-IP 仮想サーバ IP アドレスに送信します。
- ②REGISTER リクエストを受信したBIG-IP Local Traffic Manager は負荷分散処理を行い、Pool メンバのWebOTX SIP Application Server へBIG-IP 管理用 IP アドレスのVia ヘッダを追加したREGISTER リクエストを送信します。
- ③REGISTER リクエストを受信したWebOTX SIP Application Server は、To ヘッダとContact ヘッダの

組み合わせを、ローカルネットワーク内のロケーションサーバに保持します。その後、REGISTER リクエストの Contact ヘッダを BIG-IP 仮想サーバ IP アドレスに変更し、Via ヘッダを追加し、Request-URI にプロキシします。

- ④REGISTER リクエストのレスポンス信号は、Via ヘッダを元にリクエストと逆のルートをたどり、送信されます。

5.2. ローカルネットワーク発信-外部ネットワーク着信処理

ローカルネットワーク内の端末から発信し、外部ネットワーク内の端末に着信時の処理フロー、およびセッション確立後の着信側からのサブシーケントリクエストのSIPシーケンスを図 5-2 に示します。

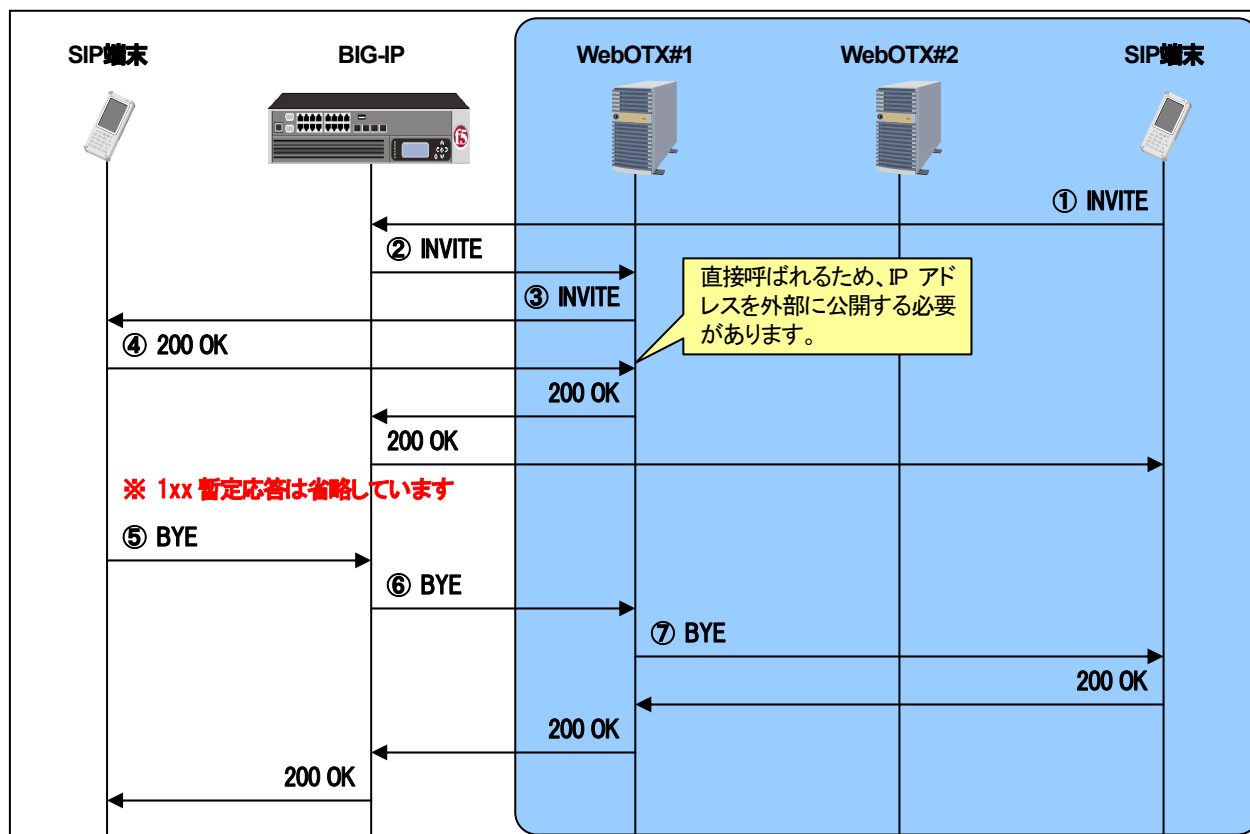


図 5-2 ローカルネットワーク発信-外部ネットワーク着信 SIP シーケンス

- ①ローカルネットワーク端末は INVITE リクエストを BIG-IP 仮想サーバ IP アドレスへ送信します。
- ②INVITE リクエストを受信した BIG-IP Local Traffic Manager は負荷分散処理を行い、Call-Id 毎に振り分け先を保持します。その後、Pool メンバの WebOTX SIP Application Server へ BIG-IP 管理用 IP アドレスの Via ヘッダ、および BIG-IP 仮想サーバ IP アドレスの Record-Route を追加した INVITE リクエストを送信します。
- ③WebOTX SIP Application Server は、ローカルネットワーク内のロケーションサーバに、To ヘッダで指定されているユーザの問い合わせを行います。問い合わせた結果、ローカルネットワークに登録されていないため、外部ネットワーク上の Proxy サーバへ、Via ヘッダを追加した INVITE リクエストをプロキシします。WebOTX SIP Application Server は Record-Route ヘッダの追加は行いません。
- ④宛先端末が送信するレスポンス信号は、Via ヘッダを元にリクエストと逆のルートを送信されます。
- ⑤外部ネットワーク端末は確立済み SIP セッションの Route 情報を元に、BIG-IP 仮想サーバ IP アドレスへサブシーケントリクエストを送信します。
- ⑥BIG-IP Local Traffic Manager は、Call-Id から INVITE リクエストをプロキシした WebOTX SIP Application Server を特定します。その後、先頭の Route ヘッダを削除し、プロキシ先の WebOTX SIP Application Server を Route ヘッダの先頭に追加、および BIG-IP 管理用 IP アドレスの Via ヘッダを

追加し、サブシーケントリクエストをプロキシします。

- ⑦Web0TX SIP Application Server は、Route ヘッダを削除し、Via ヘッダを追加後、ローカルネットワーク端末へサブシーケントリクエストをプロキシします。ローカルネットワーク端末が送信するレスポンス信号は、Via ヘッダを元にリクエストと逆のルートをたどり、送信されます。

5.3. 外部ネットワーク発信-ローカルネットワーク着信処理

外部ネットワーク内の端末から発信し、ローカルネットワーク内の端末に着信時の処理フロー、およびセッション確立後の着信側からのサブシーケントリクエストのSIPシーケンスを図 5-3に示します。

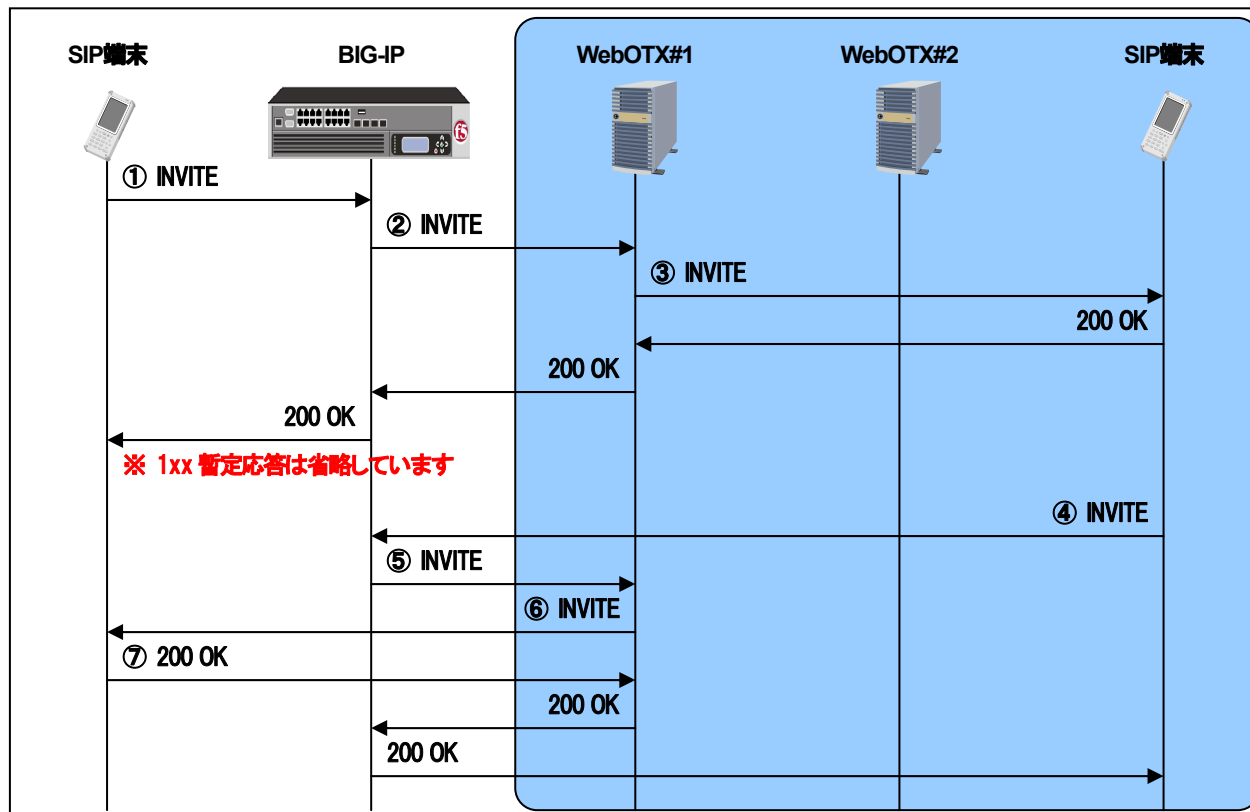


図 5-3 外部ネットワーク発信-ローカルネットワーク着信 SIP シーケンス

- ①外部ネットワーク端末は、BIG-IP 仮想サーバ IP アドレスへ INVITE リクエストを送信します。
- ②INVITE リクエストを受信した BIG-IP Local Traffic Manager は負荷分散処理を行い、Call-Id 毎に振り分け先を保持します。その後、Pool メンバの WebOTX SIP Application Server へ BIG-IP 管理用 IP アドレスの Via ヘッダ、および BIG-IP 仮想サーバ IP アドレスの Record-Route を追加した INVITE リクエストを送信します。
- ③WebOTX SIP Application Server は、ローカルネットワーク内のロケーションサーバに、To ヘッダで指定されているユーザの問い合わせを行います。問い合わせの結果、ローカルネットワークに登録されているため、ローカルネットワーク端末を特定し、Via ヘッダを追加した INVITE リクエストをプロキシします。WebOTX SIP Application Server は Record-Route ヘッダの追加は行いません。ローカルネットワーク端末が送信するレスポンス信号は、Via ヘッダを元にリクエストと逆のルートをたどり、送信されます。
- ④ローカルネットワーク端末はサブシーケントリクエストを確立済み SIP セッションの Route セットに基づき、BIG-IP 仮想サーバ IP アドレスへ送信します。
- ⑤BIG-IP Local Traffic Manager は、Call-Id から INVITE リクエストをプロキシした WebOTX SIP Application Server を特定します。その後、先頭の Route ヘッダを削除し、プロキシ先の WebOTX SIP Application Server を Route ヘッダの先頭に追加、および BIG-IP 管理用 IP アドレスの Via ヘッダを追加し、サブシーケントリクエストをプロキシします。

- ⑥Web0TX SIP Application Server は、ローカルネットワーク内のロケーションサーバに、To ヘッダで指定されているユーザの問い合わせを行います。問い合わせた結果、ローカルネットワークに登録されていないため、外部ネットワーク上の端末へ、Via ヘッダを追加した INVITE リクエストをプロキシします。また、先頭の Route ヘッダを削除します。
- ⑦外部ネットワーク端末が送信するレスポンス信号は、Via ヘッダを元にリクエストと逆のルートをたどり、送信されます。

5.4. ローカルネットワーク発信-ローカルネットワーク着信SIPシーケンス

ローカルネットワーク内の端末から発信し、ローカルネットワーク内の端末に着信時の処理フロー、およびセッション確立後の着信側からのサブシーケントリクエストのSIPシーケンスを図 5-4に示します。

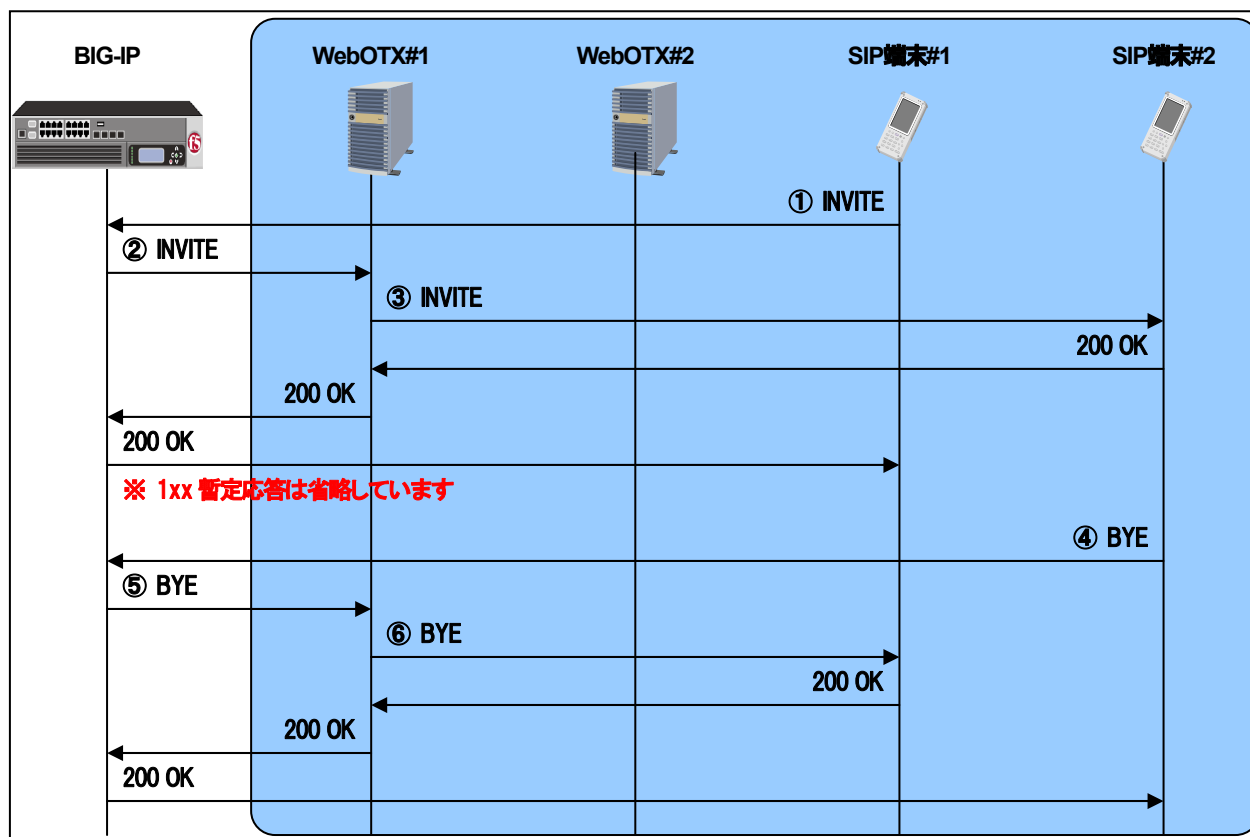


図 5-4 ローカルネットワーク発信-ローカルネットワーク着信 SIP シーケンス

- ①端末#1 は INVITE リクエストを BIG-IP 仮想サーバ IP アドレスへ送信します。
- ②INVITE リクエストを受信した BIG-IP Local Traffic Manager は負荷分散処理を行い、Call-Id 毎に振り分け先を保持します。その後、Pool メンバの WebOTX SIP Application Server へ BIG-IP 管理用 IP アドレスの Via ヘッダ、および BIG-IP 仮想サーバ IP アドレスの Record-Route を追加した INVITE リクエストを送信します。
- ③WebOTX SIP Application Server は、ローカルネットワーク内のロケーションサーバに、To ヘッダで指定されているユーザの問い合わせを行います。問い合わせた結果、ローカルネットワークに登録されているため、端末#2 を特定し、Via ヘッダを追加した INVITE リクエストをプロキシします。WebOTX SIP Application Server は Record-Route ヘッダの追加は行いません。端末#2 が送信するレスポンス信号は、Via ヘッダを元にリクエストと逆のルートをたどり、送信されます。
- ④端末#2 は確立済み SIP セッションの Route セットを元に、BIG-IP 仮想サーバ IP アドレスへ、サブシーケントリクエストを送信します。
- ⑤BIG-IP Local Traffic Manager は、Call-Id から INVITE リクエストをプロキシした WebOTX SIP Application Server を特定します。その後、先頭の Route ヘッダを削除し、プロキシ先の WebOTX SIP Application Server を Route ヘッダの先頭に追加、および BIG-IP 管理用 IP アドレスの Via ヘッダを

追加し、サブシーケントリクエストをプロキシします。

- ⑥Web0TX SIP Application Server は、Route ヘッダを削除し、Via ヘッダを追加後、端末#1 へサブシーケントリクエストをプロキシします。端末#1 が送信するレスポンス信号は、Via ヘッダを元にリクエストと逆のルートをとどり、送信されます。

6. おわりに

BIG-IP Local Traffic Manager と Web0TX SIP Application Server 間の連携システムを構築するためのガイドとしてまとめました。

本連携方式により、複数の Web0TX SIP Application server のリソースへの負荷分散やシステム負荷に応じた Web0TX SIP Application server の追加など、高いシステムの可用性、拡張性を実現することができます。

本構築ガイド作成に当たり、多大なご協力をいただきました、F5 ネットワークスジャパン社様に謝意を表します。