



# ENUM クライアント

Cisco Unified Border Element (SP Edition) は、E.164 番号マッピング (ENUM) をサポートします。

## SNMP の実装の機能履歴

リリース	変更内容
Cisco IOS XE Release 3.1S	ENUM クライアント機能が導入されました。

## 内容

- 「ENUM クライアント設定について」 (P.1017)
- 「ENUM クライアントの設定」 (P.1018)
- 「ENUM クライアントの設定例」 (P.1027)

## ENUM クライアント設定について

E.164 番号マッピング (ENUM) は、電話番号が DNS サーバで維持されるように、IP アドレスに電話番号を変換する（およびその逆の処理を行う）ための IETF 標準プロトコルです。

SBC ENUM クライアントは設定可能であり、国際電話番号の ITU 標準形式である E.164（国番号、市外局番、電話番号）を受け入れます。

ENUM クライアントは、電話番号を標準 SIP/SIP URI に変換します。これは DNS サーバによって解決され、SBC ルーティングテーブルに格納されます。現在は、IPv4 だけがサポートされています。

電話番号がコールされると、ENUM クライアントは SIP/SIP URI を DNS サーバに問い合わせます。DNS サーバは ENUM クライアントに URI を返し、ENUM クライアントは SBC のルーティングテーブルに URI を保存します。

### 宛先アドレス

着信者番号の宛先アドレスは、一般に、Request URI から取得されます。ただし、宛先アドレスはルーティングテーブルの他のヘッダー（To: ヘッダーや P-Called-Party-Id: ヘッダー）から取得されることもあります。

ENUM クライアント機能は、ヘッダーの優先順位リストを設定する機能をユーザに提供します。このリストは、To: ヘッダーや Request URI など、非必須 SIP ヘッダーで構成されることがあります。リストが設定されると、SBC はこのヘッダー リストから着信者番号の宛先アドレスを取得できます。

宛先アドレス ヘッダーは、ヘッダー フィルタ プロファイル MIB テーブルに保存されます。宛先アドレスは RFC 3261 で定義されているアドレス構文仕様に準拠する必要があります。アドレス ヘッダー リストには、最大 10 のエントリが含まれることがあります。

ENUM クライアントは最初に、Request URI を検索します。一致する着信者番号が見つからない場合、ヘッダー リストを検索します。

### 送信元アドレス

発信者番号の送信元アドレスは、一般に *From:* ヘッダーから取得されます。送信元アドレスは、次の設定を使用して変更できます。

```
header-profile <name>
  src-address
```

また、発信者番号の送信元アドレスを取得するヘッダーの優先順位リストを設定できます。このリストは、非必須 SIP ヘッダーで構成されることがあります。

送信元アドレス ヘッダーは、ヘッダー フィルタ プロファイル MIB テーブルに保存されます。送信元アドレスは RFC 3261 で定義されているアドレス構文仕様に準拠する必要があります。アドレス ヘッダー リストには、最大 10 のエントリが含まれることがあります。

### diverted-by アドレス

ENUM クライアント機能は、別の番号によって転送されるコールのヘッダーの優先順位リストから送信元番号を取得するようにサポートされています。コールが別の番号によって転送された場合、送信元アドレスは、ヘッダーの *diverted-by* リストから取得する必要があります。ユーザは、このタイプのコールを拒否するヘッダー アクションを設定することもできます。

### ヘッダー プロファイル

ユーザは、ヘッダー プロファイルを設定することで、ターゲット アドレスでアクションを実行するように設定できます。

次のアクションは、ターゲット アドレスのヘッダー プロファイルで設定できます。

- goto-table-name
- complete
- reject

SBC ENUM クライアント設定の手順については、「[ENUM クライアントの設定](#)」(P.1018) を参照してください。

SBC ENUM クライアント設定例については、「[ENUM クライアントの設定例](#)」(P.1027) を参照してください。

また、次のコマンドを使用して、SIP DNS キャッシュを設定することもできます。

- **cache-lifetime** : キャッシュされた DNS エントリのライフタイムを設定します。
- **cache-limit** : キャッシュに格納できるエントリの最大数を設定します。

## ENUM クライアントの設定

ここでは、2 種類の設定を説明します。

- 「[ENUM クライアントの設定](#)」(P.1019)
- 「[複数の ENUM エントリのコール ポリシーの設定](#)」(P.1020)

## ENUM クライアントの設定

ENUM クライアントを設定するには、次の手順を実行します。

### 手順の概要

1. **configure terminal**
2. **sbc *sbc-name***
3. **sbe**
4. **enum *enum-id***
5. **req-timeout *timeout***
6. **max-recursive-depth *number***
7. **entry *entry-name***
8. **server ipv4 *ip\_address* [*vrf vrf\_name*]**
9. **dial-plan-suffix *suffix***
10. **max-responses *number***
11. **activate**
12. **end**

### 手順の詳細

	コマンドまたはアクション	目的
ステップ1	<b>configure terminal</b>  例： Router# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ2	<b>sbc <i>sbc-name</i></b>  例： Router(config)# sbc MySBC	Cisco Unified Border Element (SP Edition) で SBC サービスを作成し、SBC コンフィギュレーション モードを開始します。
ステップ3	<b>sbe</b>  例： Router(config-sbc)# sbe	SBC の Signaling Border Element (SBE) 機能のモードを開始します。
ステップ4	<b>enum <i>enum-id</i></b>  例： Router(config-sbc-sbe)# enum 1	ENUM CUSTOMER ID 番号を割り当て、ENUM コンフィギュレーション モードを開始します。現在は、番号 1 だけが許可されます。
ステップ5	<b>req-timeout <i>timeout</i></b>  例： Router(config-sbc-sbe-enum)# req-timeout 10000	ENUM 要求タイムアウト時間を設定します。

	コマンドまたはアクション	目的
ステップ6	<code>max-recursive-depth number</code>  例： Router(config-sbc-sbe-enum)# max-recursive-depth 100	非終端の Resource Record (RR; リソース レコード) の再帰的 ENUM ルックアップの最大数を設定します。
ステップ7	<code>entry entry-name</code>  例： Router(config-sbc-sbe-enum)# entry ENUM_1	ENUM クライアント エントリ名を設定し、ENUM エントリ コンフィギュレーション モードを開始します。
ステップ8	<code>server ipv4 ip_address [vrf vrf_name]</code>  例： Router(config-sbc-sbe-enum-entry)# server ipv4 10.10.10.10 vrf VRF1	ENUM クライアントの DNS サーバの IPv4 アドレスを設定します。オプションで、VRF に DNS サーバを関連付けます。
ステップ9	<code>dial-plan-suffix suffix</code>  例： Router(config-sbc-sbe-enum-entry)# dial-plan-suffix Example.Suffix	ENUM クエリーに使用するダイヤル プラン サフィックスを設定します。
ステップ10	<code>max-responses number</code>  例： Router(config-sbc-sbe-enum)# max-responses 100	ルーティング モジュールに戻される ENUM レコードの最大数を設定します。
ステップ11	<code>activate</code>  例： Router(config-sbc-sbe-enum)# activate	ENUM クライアントをアクティブ化します。
ステップ12	<code>end</code>  例： Router(config-sbc-sbe-enum-entry)# end	コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

## 複数の ENUM エントリのコール ポリシーの設定

複数の ENUM エントリのコール ポリシーを設定するには、次の手順を実行します。

### 手順の概要

1. `configure terminal`
2. `sbc sbc-name`
3. `sbe`
4. `enum enum-id`
5. `entry (enum) entry-name`
6. `server ipv4 ip_address [vrf vrf_name]`

7. **dial-plan-suffix** *suffix*
8. **entry (enum)** *entry-name*
9. **server ipv4** *ip\_address* [**vrf** *vrf\_name*]
10. **dial-plan-suffix** *suffix*
11. **activate**
12. **exit**
13. **sip header-profile** *profile-name*
14. **dst-address**  
または  
**src-address**  
または  
**div-address**
15. **header-prio** *priority-level* **header-name** *header-name*
16. **exit**
17. **call-policy-set** *policy-set-id*
18. **first-call-routing-table** *table-name*
19. **rtg-src-adjacency-table** *table-id*
20. **entry** *entry-id*
21. **enum** *enum-id* **entry (enum)** *entry-name*
22. **action next-table** *goto-table-name*
23. **entry** *entry-id*
24. **match-adjacency** *key*
25. **enum** *enum-id* **entry (enum)** *entry-name*
26. **dst-adjacency** *target-adjacency*
27. **action complete**
28. **rtg-dst-address-table** *table-id*
29. **entry** *entry-id*
30. **match-address** *key*
31. **dst-adjacency** *target-adjacency*
32. **action complete**
33. **entry** *entry-id*
34. **match-address** *key*
35. **dst-adjacency** *target-adjacency*
36. **action complete**
37. **entry** *entry-id*
38. **match-address** *key*
39. **prefix**
40. **dst-adjacency** *target-adjacency*
41. **action complete**

42. complete

43. end

## 手順の詳細

	コマンドまたはアクション	目的
ステップ1	<code>configure terminal</code>  例： Router# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ2	<code>sbc sbc-name</code>  例： Router(config)# sbc MySBC	Cisco Unified Border Element (SP Edition) で SBC サービスを作成し、SBC コンフィギュレーション モードを開始します。
ステップ3	<code>sbe</code>  例： Router(config-sbc)# sbe	SBC の Signaling Border Element (SBE) 機能のモードを開始します。
ステップ4	<code>enum enum-id</code>  例： Router(config-sbc-sbe)# enum 1	ENUM ID 番号を割り当て、ENUM コンフィギュレーション モードを開始します。現在は、番号 1 だけが許可されません。
ステップ5	<code>entry (enum) entry-name</code>  例： Router(config-sbc-sbe-enum)# entry default-enum	デフォルトの ENUM エントリを設定し、ENUM エントリ コンフィギュレーション モードを開始します。
ステップ6	<code>server ipv4 ip_address [vrf vrf_name]</code>  例： Router(config-sbc-sbe-enum-entry)# server ipv4 10.10.10.10	ENUM クライアントの DNS サーバの IPv4 アドレスを設定します。
ステップ7	<code>dial-plan-suffix suffix</code>  例： Router(config-sbc-sbe-enum-entry)# dial-plan-suffix e164.arpa	この ENUM クエリーに使用するダイヤル プラン サフィックスを設定します。
ステップ8	<code>entry (enum) entry-name</code>  例： Router(config-sbc-sbe-enum-entry)# entry cisco-enum	別の ENUM エントリを設定し、ENUM エントリ コンフィギュレーション モードを開始します。
ステップ9	<code>server ipv4 ip_address [vrf vrf_name]</code>  例： Router(config-sbc-sbe-enum-entry)# server ipv4 10.0.0.22 vrf cisco-vrf	ENUM クライアントの DNS サーバの IPv4 アドレスを設定し、VRF に DNS サーバを関連付けます。

	コマンドまたはアクション	目的
ステップ 10	<b>dial-plan-suffix</b> <i>suffix</i>  <b>例 :</b> Router(config-sbc-sbe-enum-entry) # dial-plan-suffix cisco.com	この ENUM クエリーに使用するダイヤル プラン サフィックスを設定します。
ステップ 11	<b>activate</b>  <b>例 :</b> Router(config-sbc-sbe-enum-entry) # activate	ENUM クライアントをアクティブ化します。
ステップ 12	<b>exit</b>  <b>例 :</b> Router(config-sbc-sbe-enum) # exit	前のモードに戻ります。
ステップ 13	<b>sip header-profile</b> <i>profile-name</i>  <b>例 :</b> Router(config-sbc-sbe) # sip header-profile enum	SBE エンティティ モードでヘッダー プロファイルを設定します。
ステップ 14	<b>dst-address</b> or <b>src-address</b> or <b>div-address</b>  <b>例 :</b> Router(config-sbc-sbe-sip-hdr) # dst-address or Router(config-sbc-sbe-sip-hdr) # src-address or Router(config-sbc-sbe-sip-hdr) # div-address	宛先アドレス サブモードを開始します。 または 送信元アドレス サブモードを開始します。 または diverted-by アドレス サブモードを開始します。
ステップ 15	<b>header-prio</b> <i>priority-level</i> <b>header-name</b> <i>header-name</i>  <b>例 :</b> Router(config-sbc-sbe-sip-hdr-dst) # header-prio 1 header-name Dst_Add_Hdr_1 or Router(config-sbc-sbe-sip-hdr-src) # header-prio 1 header-name Src_Add_Hdr_1 or Router(config-sbc-sbe-sip-hdr-div) # header-prio 1 header-name Div_Add_Hdr_1	宛先アドレスが取得されるヘッダーの優先順位を設定します。 または 送信元アドレスが取得されるヘッダーの優先順位を設定します。 または diverted-by アドレスを介して取得したヘッダーの優先順位を設定します。
ステップ 16	<b>exit</b>  <b>例 :</b> Router(config-sbc-sbe-sip-hdr-src) # exit	前のモードに戻ります。

	コマンドまたはアクション	目的
ステップ 17	<code>call-policy-set</code> <i>policy-set-id</i>  例： Router(config-sbc-sbe-sip-hdr)# call-policy-set 1	新しいコール ポリシー セットを作成し、SBE ルーティング ポリシー コンフィギュレーション モードを開始します。
ステップ 18	<code>first-call-routing-table</code> <i>table-name</i>  例： Router(config-sbc-sbe-rtgpolicy)# first-call-routing-table rt1	新しいコール イベントに対してポリシーのルーティング段階を実行するとき最初に処理するポリシー テーブルの名前を設定します。
ステップ 19	<code>rtg-src-adjacency-table</code> <i>table-id</i>  例： Router(config-sbc-sbe-rtgpolicy)# rtg-src-adjacency-table rt1	既存のルーティング テーブル（この場合は rt1）のコンフィギュレーション モードを開始します。
ステップ 20	<code>entry</code> <i>entry-id</i>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable)# entry 2	ルーティング テーブルのエントリを作成します。
ステップ 21	<code>enum</code> <i>enum-id</i> <code>entry</code> ( <code>enum</code> ) <i>entry-name</i>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry)# enum 1 entry default-enum	ルーティング テーブル エントリのデフォルト ENUM エントリを設定します。
ステップ 22	<code>action next-table</code> <i>goto-table-name</i>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry)# action next-table dal	ルーティング テーブル エントリ 1 を受け取るように、アクションを設定します。
ステップ 23	<code>entry</code> <i>entry-id</i>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry)# entry 2	ルーティング テーブルのエントリを作成します。
ステップ 24	<code>match-adjacency</code> <i>key</i>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry)# match-adjacency sip2	送信元隣接に対して、エントリ 1 の照合値を設定します。この場合、送信元隣接は sip2 です。
ステップ 25	<code>enum</code> <i>enum-id</i> <code>entry</code> ( <code>enum</code> ) <i>entry-name</i>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry)# enum 1 entry cisco-enum	ルーティング テーブル エントリの ENUM エントリを設定します。



	コマンドまたはアクション	目的
ステップ 26	<code>dst-adjacency target-adjacency</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # dst-adjacency sip-proxy1	ルーティング テーブルのエントリ 2 の宛先隣接を設定します。
ステップ 27	<code>action complete</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # action complete	ルーティング テーブル エントリ 2 を受け取るように、アクションを設定します。この場合は、アクション完了です。
ステップ 28	<code>rtg-dst-address-table table-id</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # rtg-dst-address-table dal	着信者番号と照合するように、宛先アドレスを検索するルーティング テーブル (dal) を指定します。
ステップ 29	<code>entry entry-id</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable)# entry 1	ルーティング テーブルのエントリを作成します。
ステップ 30	<code>match-address key</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # match-address bob	ルーティング テーブル内のエントリ 1 の照合値を、宛先番号と照合するように設定します。
ステップ 31	<code>dst-adjacency target-adjacency</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # dst-adjacency sip-proxy2	ルーティング テーブルのエントリ 1 の宛先隣接を設定します。
ステップ 32	<code>action complete</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # action complete	ルーティング テーブル エントリ 1 を受け取るように、アクションを設定します。この場合は、アクション完了です。
ステップ 33	<code>entry entry-id</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # entry 2	ルーティング テーブルのエントリを作成します。
ステップ 34	<code>match-address key</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # match-address kate	ルーティング テーブル内のエントリ 2 の照合値を、宛先番号と照合するように設定します。

	コマンドまたはアクション	目的
ステップ 35	<code>dst-adjacency target-adjacency</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # dst-adjacency sip-proxy3	ルーティング テーブルのエントリ 2 の宛先隣接を設定します。
ステップ 36	<code>action complete</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # action complete	ルーティング テーブル エントリ 2 を受け取るように、アクションを設定します。この場合は、アクション完了です。
ステップ 37	<code>entry entry-id</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # entry 3	ルーティング テーブルのエントリを作成します。
ステップ 38	<code>match-address key</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # match-address 44	ルーティング テーブル内のエントリ 3 の照合値を、宛先番号と照合するように設定します。
ステップ 39	<code>prefix</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # prefix	このエントリの <code>match-address</code> をアドレスの先頭部分と照合するかどうかを設定します。
ステップ 40	<code>dst-adjacency target-adjacency</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # dst-adjacency sip-proxy4	ルーティング テーブルのエントリ 3 の宛先隣接を設定します。
ステップ 41	<code>action complete</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # action complete	ルーティング テーブル エントリ 3 を受け取るように、アクションを設定します。この場合は、アクション完了です。
ステップ 42	<code>complete</code>  例： Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # complete	フルセットのコミット後にコール ポリシー セットを完了します。
ステップ 43	<code>end</code>  例： Router(config-sbc-sbe-rtgpolicy)# end	コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

# ENUM クライアントの設定例

## 例 1：ENUM クライアント

ENUM クライアントを設定するには、次の手順を実行します。

```
Router# configure terminal
Router(config)# sbc MySBC
Router(config-sbc)# sbe
Router(config-sbc-sbe)# enum 1
Router(config-sbc-sbe-enum)# req-timeout 10000
Router(config-sbc-sbe-enum)# max-recursive-depth 100
Router(config-sbc-sbe-enum)# entry ENUM_1
Router(config-sbc-sbe-enum-entry)# server ipv4 10.10.10.10 vrf VRF1
Router(config-sbc-sbe-enum-entry)# dial-plan-suffix Example.Suffix
Router(config-sbc-sbe-enum)# max-responses 100
Router(config-sbc-sbe-enum)# activate
Router(config-sbc-sbe-enum-entry)# end
```

## 例 2：複数の ENUM エントリのコール ポリシー

複数の ENUM エントリのコール ポリシーを設定するには、次の手順を実行します。

```
Router# configure terminal
Router(config)# sbc mysbc
Router(config-sbc)# sbe
Router(config-sbc-sbe)# enum 1
Router(config-sbc-sbe-enum)# entry default-enum
Router(config-sbc-sbe-enum-entry)# server ipv4 192.168.10.1
Router(config-sbc-sbe-enum-entry)# dial-plan-suffix e164.arpa
Router(config-sbc-sbe-enum-entry)# entry cisco-enum
Router(config-sbc-sbe-enum-entry)# server ipv4 10.0.0.22 vrf cisco-vrf
Router(config-sbc-sbe-enum-entry)# dial-plan-suffix cisco.com
Router(config-sbc-sbe-enum-entry)# activate
Router(config-sbc-sbe-enum)# exit

Router(config-sbc-sbe)# sip header-profile enum
Router(config-sbc-sbe-sip-hdr) dst-address
Router(config-sbc-sbe-sip-hdr-dst)# header-prio 1 header-name Dst_Add_Hdr_1
Router(config-sbc-sbe-sip-hdr-dst)# header-prio 2 header-name Dst_Add_Hdr_2
Router(config-sbc-sbe-sip-hdr-dst)# exit
or
Router(config-sbc-sbe)# sip header-profile enum
Router(config-sbc-sbe-sip-hdr) src-address
Router(config-sbc-sbe-sip-hdr-src)# header-prio 1 header-name Src_Add_Hdr_1
Router(config-sbc-sbe-sip-hdr-src)# header-prio 2 header-name Src_Add_Hdr_2
Router(config-sbc-sbe-sip-hdr-src)# exit
or
Router(config-sbc-sbe)# sip header-profile enum
Router(config-sbc-sbe-sip-hdr) div-address
Router(config-sbc-sbe-sip-hdr-div)# header-prio 1 header-name Div_Add_Hdr_1
Router(config-sbc-sbe-sip-hdr-div)# header-prio 2 header-name Div_Add_Hdr_2
Router(config-sbc-sbe-sip-hdr-div)# exit

Router(config-sbc-sbe-sip-hdr)# call-policy-set 1
Router(config-sbc-sbe-rtgpolicy)# first-call-routing-table rtl
Router(config-sbc-sbe-rtgpolicy)# rtg-src-adjacency-table rtl

Router(config-sbc-sbe-rtgpolicy-rtgtable)# entry 1
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry)# match-adjacency sip2
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry)# enum 1 entry default-enum
```

```
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # action next-table da1

Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # entry 2
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # match-adjacency sip2
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # enum 1 entry cisco-enum
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # dst-adjacency sip-proxy1
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # action complete
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # rtg-dst-address-table da1

Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # entry 1
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # match-address bob
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # dst-adjacency sip-proxy2
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # action complete

Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # entry 2
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # match-address kate
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # dst-adjacency sip-proxy3
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # action complete

Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # entry 3
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # match-address 44
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # prefix
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # dst-adjacency sip-proxy4
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # action complete
Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # complete
Router(config-sbc-sbe-rtgpolicy) # end
Router#
```