



ENUM Client

Cisco Unified Border Element (SP Edition) supports E.164 Number Mapping (ENUM).

Feature History for Implementing SNMP

| Release | Modification |
|---------------------------|-------------------------------------|
| Cisco IOS XE Release 3.1S | ENUM Client Feature was introduced. |

Contents

- [Information about ENUM Client Configuration, page 53-1](#)
- [Configuring ENUM Client, page 53-3](#)
- [Configuration Examples of ENUM Client Configuration, page 53-11](#)

Information about ENUM Client Configuration

E.164 Number Mapping (ENUM) is an IETF standard protocol for converting telephone numbers into IP addresses (and vice versa), so that the telephone numbers can be maintained by a DNS server.

The SBC ENUM client is configurable and accepts the ITU standard format for international telephone numbers, E.164: country code, area code, phone number.

The ENUM client translates telephone numbers into standard sip/sips URIs that are resolved by a DNS server and then stored in an SBC routing table. Currently, only IPv4 is supported.

When a telephone number is called, the ENUM client queries the DNS server for a sip/sips URI. The DNS server returns the URI to the ENUM client, and the ENUM client stores the URI in an SBC routing table.

Destination Address

The destination address of a called number is typically derived from the Request URI. However, the destination address may also be derived from other headers in the routing table, such as the *To:* header or the *P-Called-Party-ID:* header.

The ENUM Client feature provides the user with the ability to configure a prioritized list of headers. This list may consist of any non-essential SIP headers, including the *To:* header and the Request URI. Once the list is configured, SBC can derive destination addresses for called numbers from this list of headers.

Destination address headers are stored in the header filter profile MIB table. Destination addresses must conform to the address syntax specification defined in RFC 3261. An address header list may contain a maximum to 10 entries.

The ENUM Client first searches the Request URI. If it does not find a match for the called number, it then searches the header list.

Source Address

The source address of a calling party number is typically derived from the *From:* header. The source addresses can be modified using the following configuration.

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

You can also configure a prioritized list of headers from which the source address for a calling number is derived. This list may consist of any non-essential SIP headers.

Source address headers are stored in the header filter profile MIB table. Source addresses must conform to the address syntax specification defined in RFC 3261. An address header list may contain a maximum to 10 entries.

Diverted-by Address

The ENUM Client feature also provides support for deriving the source number from a prioritized list of headers for calls which have been diverted by another number. If a call has been diverted by another number, the source address must be derived from the diverted-by list of headers. Users can also configure a header action to reject these types of calls.

Header Profiles

The user can configure actions to be performed on a target address by configuring a header profile.

The following actions can be configured in a header profile for a target address:

- goto-table-name
- complete
- reject

For the SBC ENUM client configuration steps, see the [?\\$paranum>Configuring ENUM Client? section on page 53-3](#).

For an example of SBC ENUM client configuration see the [?\\$paranum>Configuration Examples of ENUM Client Configuration? section on page 53-11](#).

Additionally, you can also configure the SIP DNS cache, using the following commands:

- **cache lifetime**—configures the lifetime of a cached DNS entry.
- **cache limit**—configures the maximum number of entries that are permitted in the cache

Configuring ENUM Client

The sections presents two configurations:

- [Configuring ENUM Client, page 53-3](#)
- [Configuring a Call Policy for Multiple ENUM Entries, page 53-4](#)

Configuring ENUM Client

Use the following procedure to configure and ENUM client:

SUMMARY STEPS

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**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|--|
| Step 1 | configure terminal Example: Router# configure terminal | Enters global configuration mode. |
| Step 2 | sbc <i>sbc-name</i> Example: Router(config)# sbc MySBC | Creates the SBC service on Cisco Unified Border Element (SP Edition) and enters into SBC configuration mode. |
| Step 3 | sbe Example: Router(config-sbc)# sbe | Enters the mode of the signaling border element (SBE) function of the SBC. |

| | Command or Action | Purpose |
|---------|--|--|
| Step 4 | enum <i>enum-id</i> Example: Router(config-sbc-sbe)# enum 1 | Assigns the ENUM CUSTOMER ID number and enters ENUM configuration mode. Currently, only the number 1 is allowed. |
| Step 5 | req-timeout <i>timeout</i> Example: Router(config-sbc-sbe-enum)# req-timeout 10000 | Configures the ENUM request timeout period. |
| Step 6 | max-recursive-depth <i>number</i> Example: Router(config-sbc-sbe-enum)# max-recursive-depth 100 | Configures the maximum number of recursive ENUM look-ups for non-terminal Resource Records (RR). |
| Step 7 | entry <i>entry-name</i> Example: Router(config-sbc-sbe-enum)# entry ENUM_1 | Configures the ENUM Client entry name and enter the ENUM entry configuration mode. |
| Step 8 | server ipv4 <i>ip_address</i> [vrf <i>vrf_name</i>] Example: Router(config-sbc-sbe-enum-entry)# server ipv4 10.10.10.10 vrf VRF1 | Configures the IPv4 address of a DNS server for ENUM Client and optionally associates the DNS server to a VRF. |
| Step 9 | dial-plan-suffix <i>suffix</i> Example: Router(config-sbc-sbe-enum-entry)# dial-plan-suffix Example.Suffix | Configures the dial plan suffix used for the ENUM query. |
| Step 10 | max-responses <i>number</i> Example: Router(config-sbc-sbe-enum)# max-responses 100 | Configures the maximum number of ENUM records returned to the routing module. |
| Step 11 | activate Example: Router(config-sbc-sbe-enum)# activate | Activates ENUM Client. |
| Step 12 | end Example: Router(config-sbc-sbe-enum-entry)# end | Exits configuration mode and returns to privileged EXEC mode. |

Configuring a Call Policy for Multiple ENUM Entries

Use the following procedure to configure a call policy for multiple ENUM entries:

SUMMARY STEPS

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**
or
src-address
or
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**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|--|
| Step 1 | configure terminal Example: Router# configure terminal | Enters global configuration mode. |
| Step 2 | sbc <i>sbc-name</i> Example: Router(config)# sbc MySBC | Creates the SBC service on Cisco Unified Border Element (SP Edition) and enters into SBC configuration mode. |
| Step 3 | sbe Example: Router(config-sbc)# sbe | Enters the mode of the signaling border element (SBE) function of the SBC. |
| Step 4 | enum <i>enum-id</i> Example: Router(config-sbc-sbe)# enum 1 | Assigns the ENUM ID number and enters ENUM configuration mode. Currently, only the number 1 is allowed. |
| Step 5 | entry (enum) <i>entry-name</i> Example: Router(config-sbc-sbe-enum)# entry default-enum | Configures the default ENUM entry and enters ENUM entry configuration mode. |
| Step 6 | server ipv4 <i>ip_address</i> [vrf <i>vrf_name</i>] Example: Router(config-sbc-sbe-enum-entry)# server ipv4 10.10.10.10 | Configures the IPv4 address of a DNS server for the ENUM Client. |

| | Command or Action | Purpose |
|---------|--|--|
| Step 7 | dial-plan-suffix <i>suffix</i> Example: Router(config-sbc-sbe-enum-entry)# dial-plan-suffix e164.arpa | Configures the dial plan suffix used for this ENUM query. |
| Step 8 | entry (enum) <i>entry-name</i> Example: Router(config-sbc-sbe-enum-entry)# entry cisco-enum | Configures another ENUM entry and enters ENUM entry configuration mode. |
| Step 9 | server ipv4 <i>ip_address</i> [vrf <i>vrf_name</i>] Example: Router(config-sbc-sbe-enum-entry)# server ipv4 10.0.0.22 vrf cisco-vrf | Configures the IPv4 address of a DNS server for ENUM Client and associates the DNS server to a VRF. |
| Step 10 | dial-plan-suffix <i>suffix</i> Example: Router(config-sbc-sbe-enum-entry)# dial-plan-suffix cisco.com | Configures the dial plan suffix used for this ENUM query. |
| Step 11 | activate Example: Router(config-sbc-sbe-enum-entry)# activate | Activates the ENUM client. |
| Step 12 | exit Example: Router(config-sbc-sbe-enum)# exit | Exits to the previous mode. |
| Step 13 | sip header-profile <i>profile-name</i> Example: Router(config-sbc-sbe)# sip header-profile enum | Configures a header profile in the mode of an SBE entity. |
| Step 14 | dst-address or src-address or div-address Example: 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 | Enters destination address submode. or Enters source address submode. or Enters diverted-by address submode. |

| | Command or Action | Purpose |
|---------|--|--|
| Step 15 | <p>header-prio <i>priority-level</i> header-name <i>header-name</i></p> <p>Example: 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</p> | <p>Configures the priority of the header from which the destination address is derived.</p> <p>or</p> <p>Configures the priority of the header from which the source address is derived.</p> <p>or</p> <p>Configures the priority of the header from which the diverted-by address is derived.</p> |
| Step 16 | <p>exit</p> <p>Example: Router(config-sbc-sbe-sip-hdr-src)# exit</p> | Exits to the previous mode. |
| Step 17 | <p>call-policy-set <i>policy-set-id</i></p> <p>Example: Router(config-sbc-sbe-sip-hdr)# call-policy-set 1</p> | Creates a new call policy set and enters SBE routing policy configuration mode. |
| Step 18 | <p>first-call-routing-table <i>table-name</i></p> <p>Example: Router(config-sbc-sbe-rtgpolicy)# first-call-routing-table rt1</p> | Configures the name of the first policy table to process when performing the routing stage of policy for new-call events. |
| Step 19 | <p>rtg-src-adjacency-table <i>table-id</i></p> <p>Example: Router(config-sbc-sbe-rtgpolicy)# rtg-src-adjacency-table rt1</p> | Enters the configuration mode of the existing routing table, in this case, rt1. |
| Step 20 | <p>entry <i>entry-id</i></p> <p>Example: Router(config-sbc-sbe-rtgpolicy-rtgtable)# entry 2</p> | Creates an entry in the routing table. |
| Step 21 | <p>enum <i>enum-id</i> entry (enum) <i>entry-name</i></p> <p>Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # enum 1 entry default-enum</p> | Configures the default ENUM entry for the routing table entry. |
| Step 22 | <p>action next-table <i>goto-table-name</i></p> <p>Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # action next-table dal</p> | Configures the action to take on routing table entry 1. |

| | Command or Action | Purpose |
|---------|---|--|
| Step 23 | entry <i>entry-id</i> Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # entry 2 | Creates an entry in the routing table. |
| Step 24 | match-adjacency <i>key</i> Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # match-adjacency sip2 | Configures the match value for entry 1 against a source adjacency. In this case, the source adjacency is sip2. |
| Step 25 | enum <i>enum-id</i> entry (enum) <i>entry-name</i> Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # enum 1 entry cisco-enum | Configures an ENUM entry for the routing table entry. |
| Step 26 | dst-adjacency <i>target-adjacency</i> Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # dst-adjacency sip-proxy1 | Configures the destination adjacency for entry 2 in table routing table. |
| Step 27 | action complete Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # action complete | Configures the action to take on routing table entry 2. In this case, the action is complete. |
| Step 28 | rtg-dst-address-table <i>table-id</i> Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # rtg-dst-address-table da1 | Specifies the routing table (da1) that is searched for destination addresses to match called numbers. |
| Step 29 | entry <i>entry-id</i> Example: Router(config-sbc-sbe-rtgpolicy-rtgtable)# entry 1 | Creates an entry in the routing table. |
| Step 30 | match-address <i>key</i> Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # match-address bob | Configures the match value for entry 1 in the routing table, to match against a destination called number. |
| Step 31 | dst-adjacency <i>target-adjacency</i> Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # dst-adjacency sip-proxy2 | Configures the destination adjacency for entry 1 in table routing table. |

| | Command or Action | Purpose |
|---------|---|--|
| Step 32 | action complete Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # action complete | Configures the action to take on routing table entry 1. In this case, the action is complete. |
| Step 33 | entry entry-id Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # entry 2 | Creates an entry in the routing table. |
| Step 34 | match-address key Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # match-address kate | Configures the match value for entry 2 in the routing table, to match against a destination called number. |
| Step 35 | dst-adjacency target-adjacency Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # dst-adjacency sip-proxy3 | Configures the destination adjacency for entry 2 in table routing table. |
| Step 36 | action complete Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # action complete | Configures the action to take on routing table entry 2. In this case, the action is complete. |
| Step 37 | entry entry-id Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # entry 3 | Creates an entry in the routing table. |
| Step 38 | match-address key Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # match-address 44 | Configures the match value for entry 3 in the routing table, to match against a destination called number. |
| Step 39 | prefix Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # prefix | Configures whether the match-address of this entry matches the start of the address. |
| Step 40 | dst-adjacency target-adjacency Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # dst-adjacency sip-proxy4 | Configures the destination adjacency for entry 3 in table routing table. |

| | Command or Action | Purpose |
|---------|---|---|
| Step 41 | action complete Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # action complete | Configures the action to take on routing table entry 3. In this case, the action is complete. |
| Step 42 | complete Example: Router(config-sbc-sbe-rtgpolicy-rtgtable-entry) # complete | Completes the call-policy set after committing the full set. |
| Step 43 | end Example: Router(config-sbc-sbe-rtgpolicy)# end | Exits configuration mode and returns to privileged EXEC mode. |

Configuration Examples of ENUM Client Configuration

Example 1: ENUM Client

Use the following procedure to configure an ENUM Client:

```
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
```

Example 2: Call Policy for Multiple ENUM Entries

Use the following procedure to configure a call policy for multiple ENUM entries:

```
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 rt1
Router(config-sbc-sbe-rtgpolicy)# rtg-src-adjacency-table rt1

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#

```