Custom Reference Data Configuration

Logical APN List

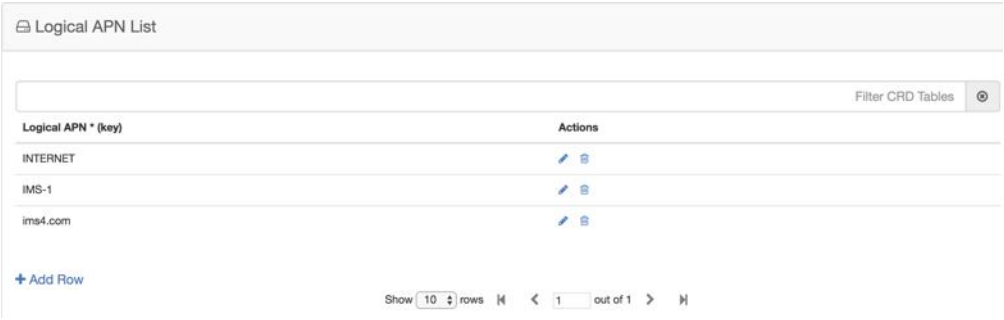
The logical APN feature allows multiple users to access different physical target networks through a shared APN access point. The logical APN feature reduces the amount of APN provisioning required by consolidating access all real APNs through a single virtual APN. Therefore, only the virtual APN needs to be provisioned at Control Centre, instead of each of the real APNs to be reached.

For details on System ID, refer to Peer Routing, on page 5.

For details on Peer Group, refer to Peer Group Mapping and Peer Group SRK Mapping.

An example configuration is shown below:

Figure 1: Logical APN List



APN Mapping Table

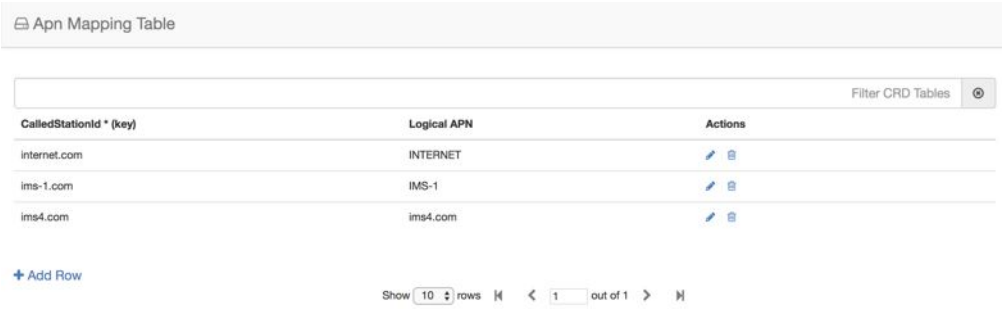
The APN consists of two parts which are as follows:

- The APN Network Identifier. This part of the APN is mandatory.
- The APN Operator Identifier. This part of the APN is optional.

The actual APN of any interface is filled-in with Called-Station-Id AVP. This table keeps a mapping of actual APNs and logical APNs configured in the logical APN list.

The following is an example configuration:

Figure 2: APN Mapping Table



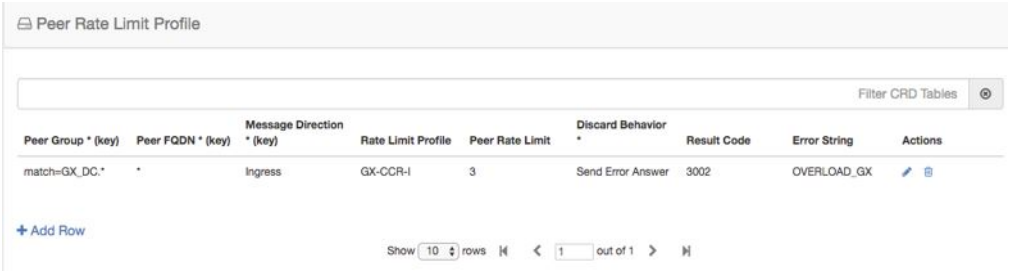
The Called-Station-Id input is case insensitive where it stores all the values in lower case. It converts the upper case entry to a lower case value and checks for a duplicate entry. If the input APN contains any duplicate value, it rejects the value with an error message.

For example, if the input value is `IMS.COM`, it stores the value as `ims.com`.

Peer Rate Limit Profile

CPS vDRA can rate limit traffic coming from and going towards a particular peer. This can work for both Ingress and Egress traffic. User needs to define the peer group, FQDN, traffic direction and the CPS vDRA behavior, whether to silently drop or send error message. User can also define the error code and the error message when error responses need to be sent back.

Figure 3: Peer Rate Limit Profile



Peer Access Control List

You can use the Peer Access Control List to specify the list of peers (by realm, FQDN, application ID, or Source-IP) that can establish peer connections to vDRA.

Peers that are not listed with realm or host in the CRD are allowed to establish peer connections by default.

Specify the following parameters:

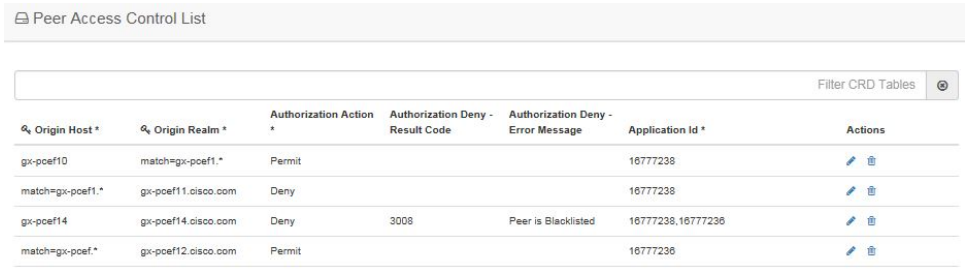
The key fields are Origin Host and Origin realm, hence it is possible to have only one row for each unique pair.

- Origin Host - Diameter identity or FQDN(host) of the client either in full or as a regular expression
- Origin Realm - Diameter Identity or realm of the client either in full or as a regular expression
- Authorization Action: Specifies whether the incoming client connection is allowed or denied.
- Authorization Deny - Result Code: Configurable result code. If not configured, the default value of 3010 (Unknown Application) or 3007 (Unsupported Application) is sent. Applicable only when the Authorization action is set to “Deny”
- Authorization Deny - Error Message: Configurable Message. If not configured default values are Unknown Peer or Unsupported Application.

  Applicable only when the Authorization action is set to “Deny”
- Application ID: single, comma-separated, or regular expression.

If the peer connection is rejected due to mismatch of Applications, customized result-code / error messages are not applicable in this case.

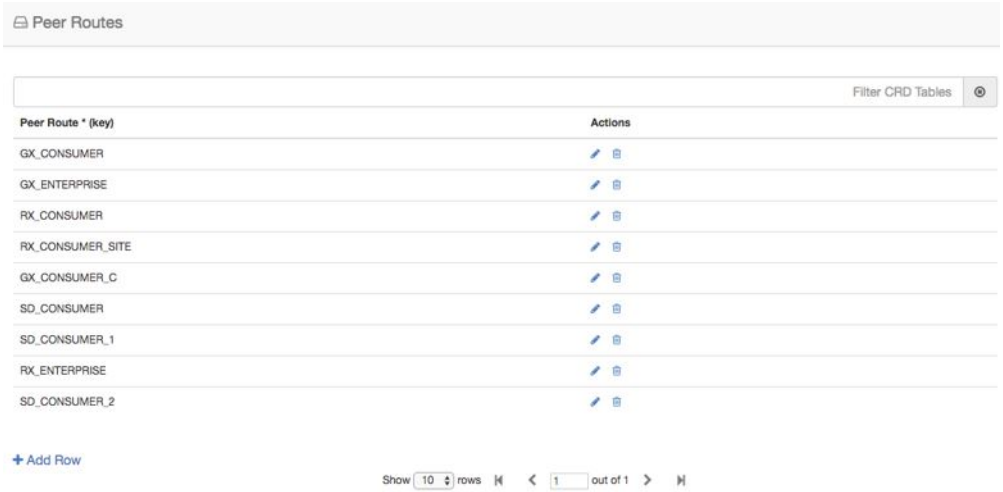
Figure 4: Peer Access Control List



Peer Routes

Request forwarding is done using Peer Routes to discover peers. These routes are different for different interfaces. There can be multiple peer routes for a particular interface.

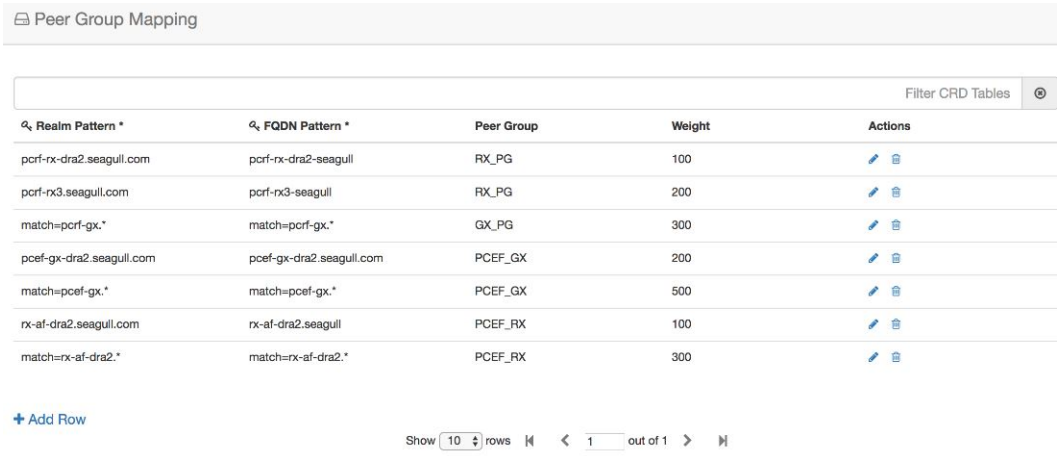
Figure 5: Peer Routes



Peer Group Mapping

One or more peers are combined into single peer group based on their realms patterns and FQDN patterns. Peer groups have respective peer routes.

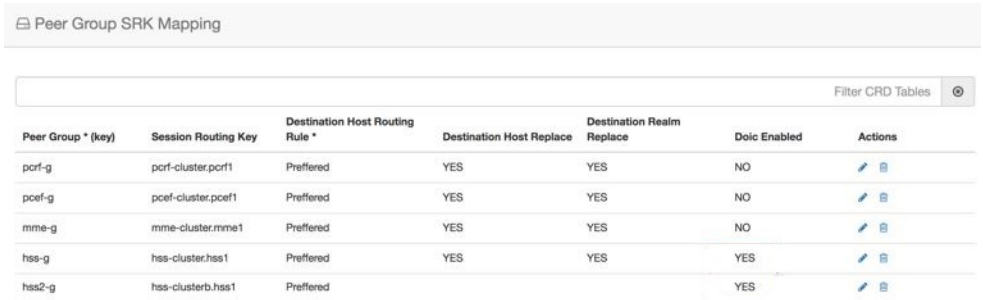
Figure 6: Peer Group Mapping



Peer Group SRK Mapping

All the peer groups consisting of one or more peers are listed in this table. Also various features like Session Key Routing or Destination Host Routing can be configured as Only, Never, Preferred depending upon the need. Use the DOIC Enabled column (YES/NO) to enable or disable Diameter Overload Indication Conveyance (DOIC). This option is used to throttle or divert Diameter requests towards PCRF, HSS, AAA, and OCS servers based on reporting of overloaded conditions.

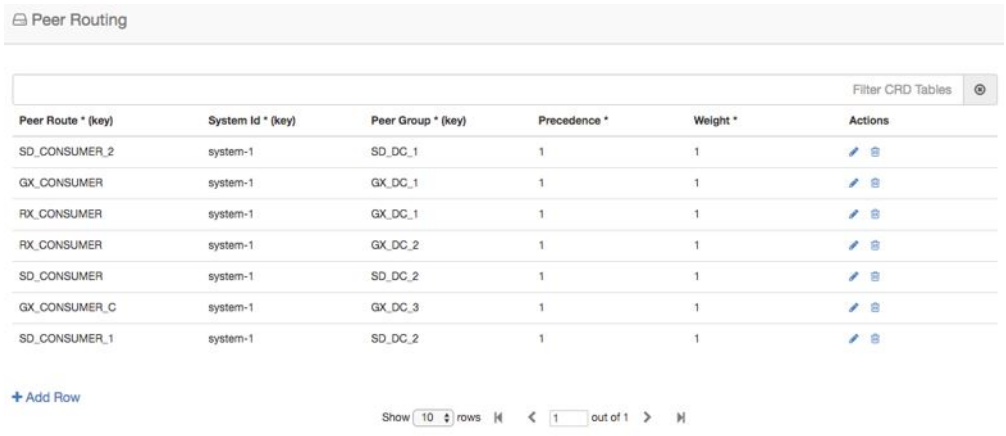
Figure 7: Peer Group SRK Mapping



Peer Routing

This table consists of a mapping of Peer Groups to Peer Routes on a particular CPS vDRA. It also has precedence and weight columns which play a vital role in load balancing behavior of CPS vDRA.

Figure 8: Peer Routing



Binding Key Profile

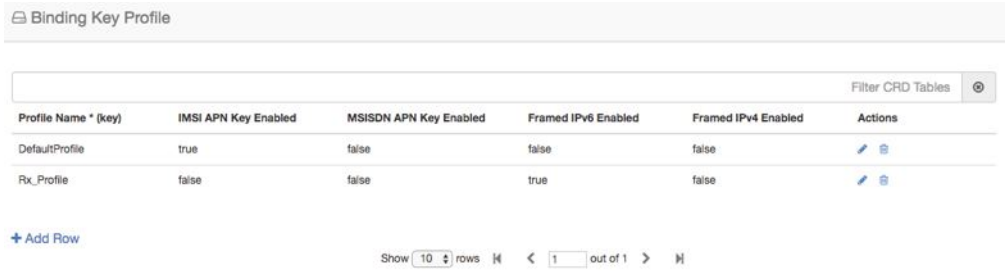
Important For routing to work in DRA, user must configure **AppId Key Profile Mapping** and **Binding Key Profile** tables.

The available fields are Boolean fields and can be edited by selecting the check boxes.

Note It is expected a minimum of one row to be configured with the value “DefaultProfile”. This will be used in case there is nothing configured for an application Id. For this “DefaultProfile”, “imsiAPN” and “FramedIPv6Prefix” should be enabled.

Note The field **MSISDN APN Key Enabled** is a place holder only. Modifying this field will not have an effect on the application behavior.

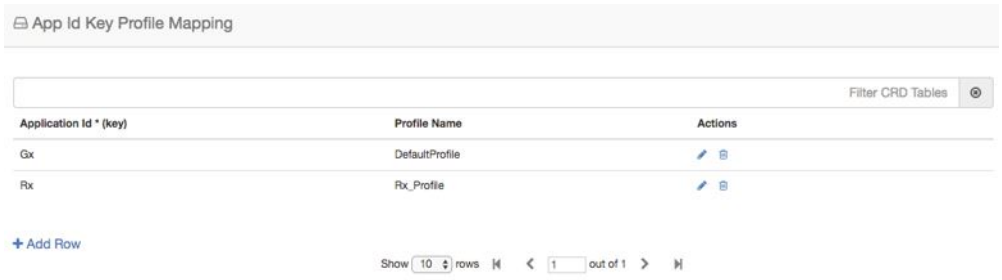
Figure 9: Binding Key Profile



AppId Key Profile Mapping

Important For routing to work in CPS vDRA, you must configure **AppId Key Profile Mapping** and **Binding Key Profile** tables.

Figure 10: AppId Key Profile Mapping



The Binding Key Profile column is tied to the Profile Name column from the previous CRD and takes the available Profile Name in the system.

There are two application Identifiers that have been provisioned in the system which are Gx and Rx and can be tied to the same or different Bind Key Profile as the case may be.

Message Class Profile

To determine the abatement action from the DOIC Profile table (for throttling or diverting Diameter requests), you require a Message class. You can query the Message class from the Message Class Profile table.

The Message Class Profile table takes inputs such as Ingress Peer Group, Application Id, Command Code, Message/Request Type and provides the Condition Profile and Message Class. Message Class can be one of P0, P1, P2, P3, P4.

Figure 11: Message Class Profile

Message Retry Profile

CPS vDRA supports configurable retries, so that the specific behavior of CPS vDRA in congestion scenarios can be configured.

Configurable retry mechanism (i.e., number of retries) per:

- Application ID
- Peer Group
- Answer Timeout error occurred
- Error Result Code of Response

This should be in the form of a CRD and applied to a peer group. The user can use the SRK peers to select an alternate peer.

If all SRK peers fail, the user should use one alternate CPS vDRA if it connects to the SRK. If the SRK matches exactly, CPS vDRA would look for the second label match of SRK like clusterb.dc1 and clusterc.dc1 and retry the message to other peer group.

Figure 12: Message Retry Profile - Control Center

Message Retry Profile

Filter CRD Tables

Peer Group * (key)	Application Id * (key)	Command Code * (key)	Result Code * (key)	Experimental RC	Number Of Retries	Actions
GX_DC_1	16777238	272	3003	false	6	
GX_DC_1	16777238	272	3004	false	6	
GX_DC_2	16777238	272	3003	false	6	
GX_DC_2	16777238	272	3004	false	6	
match=SD_DC.*	16777303	8388637	3004	false	6	
RX_DC_1	16777236	258	7000	false	6	
GX_DC_2	16777238	272	7000	false	6	

+ Add Row

Show 10 rows 1 out of 1

Wild card match is supported for Peer Group, Application Id, Command Code, Result Code columns. For example, 300.* supports all RC starting with 300.

- * is supported to allow all RC.
- * is supported for all peer groups.
- Match = GX_DC_.* is supported for groups starting with GX_DC

RC = 7000 is interpreted as retry for timeout.

Experimental result code is for future purposes and value in that column has no effect on retry processing.

Note **Best Match** check box needs to be checked in Policy Builder if you want to use the wildcard feature.

Refer to Message Retry Profile for configuration in **Search Table Group**.

Peer Group Answer Timeout

CPS vDRA support for the following use cases:

1. Configurable answer timeout for initial try and subsequent retries for the following parameters:
   - Application ID
   - Peer Group (to which request is sent)
   - Command code (to enable different timeouts for different Diameter commands)
   - Timeout value (in milliseconds)

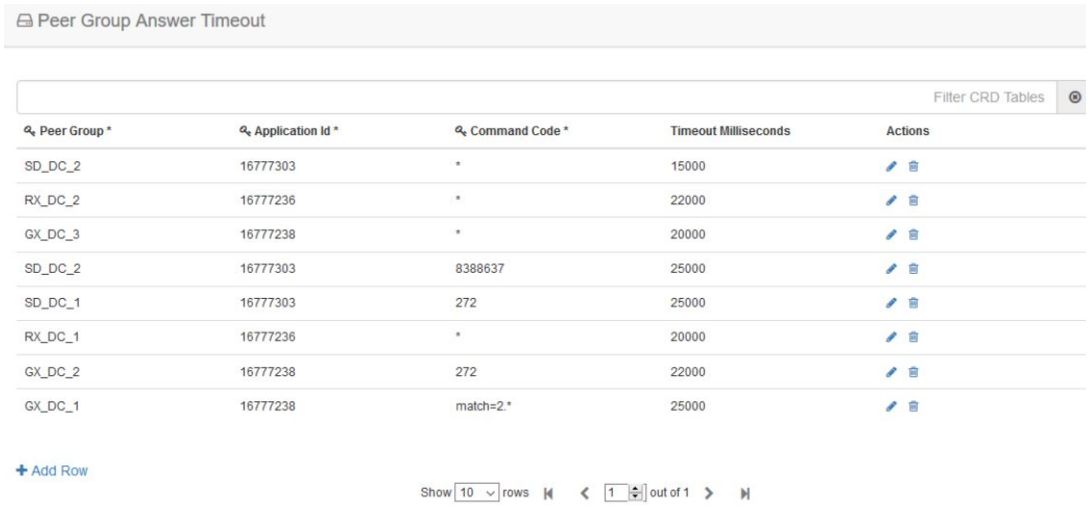
2. Default value if unspecified is 1700 milliseconds.

Peer group answer timeout is applicable for every message routed using:

- Destination host routing
- SRK routing
- Table driven routing

Sample peer group answer timeout is shown below:

Figure 13: Peer Group Answer Timeout



Wild card match is supported for application_id, peer_group, command code. * indicates all application_id, peer_group.

The following rules have been applied for answer timeout:

- Default timeout for any message routed from CPS vDRA is 1700 ms.
- In case of retry, if an alternate group is chosen for routing, corresponding timeout for the peer group is applied.

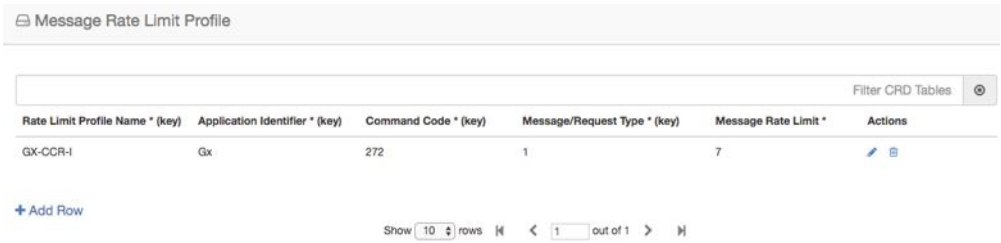
For Policy Builder related configuration, refer to Peer Group Answer Timeout.

Message Rate Limit Profile

Further to peer level rate limit, CPS vDRA provides the granularity of limiting diameter traffic at message level for each peer. Message level rate limit always works in conjunction with peer level rate limit and is an additional control in peer level rate limit configuration. Since message level rate limit works in conjunction with peer level rate limit, all the fields specified for peer level rate limit are applicable to message level rate limit.

Message Rate Limit Profile table is used to get the condition for such rate limiting. User can define the type of message, command code and the application for which the limiting has to be implemented.

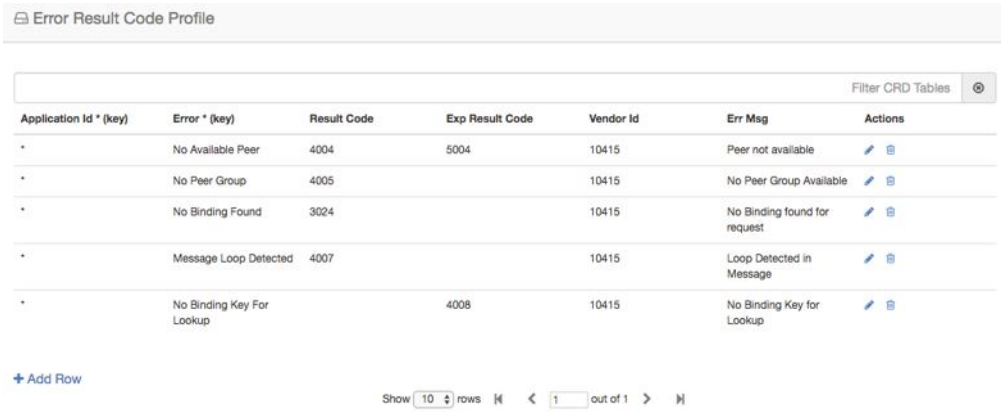
Figure 14: Message Rate Limit Profile



Error Result Code Profile

Sample CRD data looks like this:

Figure 15: Error Result Code Profile



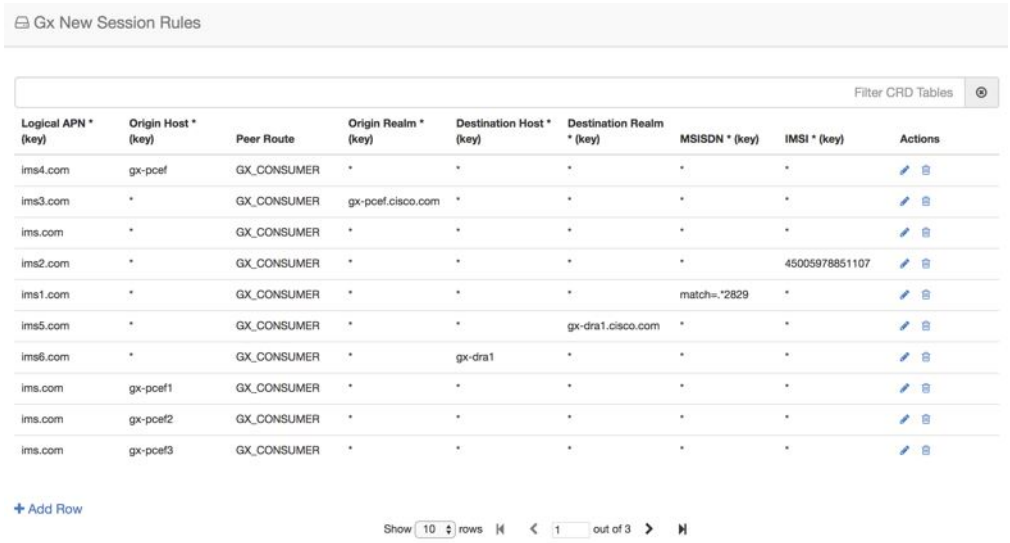
- For any CPS vDRA error or message timeout, CPS vDRAhas the ability to map the error to a Result-Code value and an error message string for the Error-Message AVP.
- Errors include things like "binding not found", "message timeout", "no peer connections".
- The Result Code value is sent in the Result-Code AVP in the response.
- The error message string is sent in the Error-Message AVP in the response.
- When both Result Code and Exp Result Code are configured in this table, Result Code will take precedence. In case Result Code is not configured in this table, Exp Result Code will be sent with Vendor-ID.

Gx New Session Rules

Gx New Session Rules table is used by CPS vDRA when performing Table Driven routing. CPS vDRA could derive the "Peer Route" from this table, when the incoming message has no destination host to be routed to. From peer route, CPS vDRA derives further route where the request could be sent. This table supports both wildcard and exact match for the various parameters. The "Peer Route" used in this table should be defined

in "Peer Routes" table. Here an example for Gx New Session Rules is provided. Similar tables can be created for Rx or Sd.

Figure 16: Gx New Session Rules



Range Based Routing

CPS vDRA provides range-based routing based on MSISDN and IMSI values so that Diameter requests are routed to the correct HSS or AAA server. Range-based routing occurs if the destination-host routing, binding-based routing and SLF-based routing fails.

- vDRA checks whether the primary lookup type is IMSI or MSISDN and also checks whether the IMSI/MSISDN value present in the request matches against the range configured in CRD.
- The primary lookup type is evaluated first and if it fails, the secondary lookup type is evaluated.
- If primary lookup type evaluation fails and if the secondary lookup type is not configured, the request is routed with table-driven routing (if configured).
- If both the primary lookup type credential and the secondary lookup type evaluation fail, the request is rejected or routed with table driven routing (if configured).

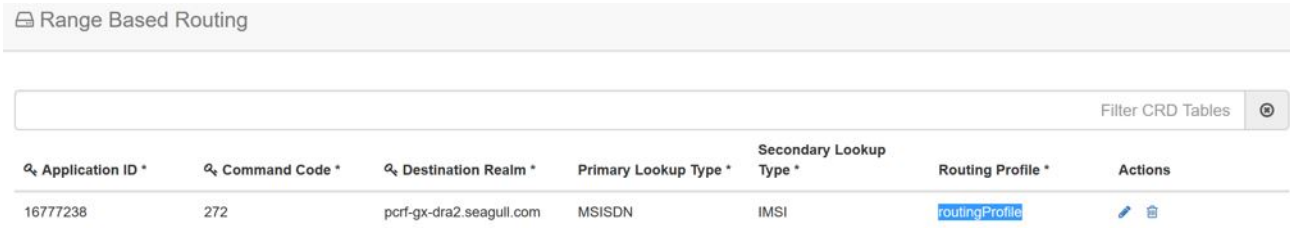
vDRA matches the request against the Range Based Routing table and based on the result of the credential match, SRK routing is initiated.

Table 1: Range Based Routing

Field	Description	Value
Application Id (input)	The diameter application of the message received	Integer value of the application id
Command Code (input)	The message command code	Integer value of the command code

Field	Description	Value
Destination Realm (input)	The destination realm in the message	String value of destination realm
Primary Lookup Type (input)	Primary lookup type for range based routing	IMSI or MSISDN
Secondary Lookup Type (input)	Secondary lookup type for range based routing	NONE or IMSI or MSISDN
Routing Profile (output)	Routing profile	Any string value. (Should match the routing profile in either or both the IMSI and MSISDN range CRD for a successful match).

Figure 17: Range Based Routing



IMSI Range

The IMSI Range is used in range-based routing to configure the range of IMSI values.

Table 2: IMSI Range

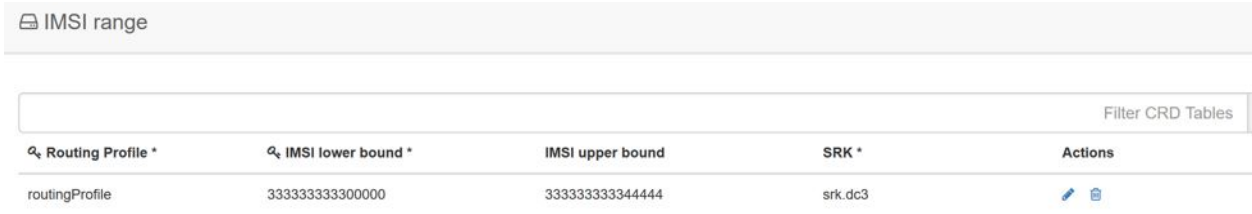
Field	Description	Value
Routing Profile (input)	The routing profile name	Any string value
IMSI lower bound (input)	The lower bound for the IMSI value	For a numeric range, enter the IMSI value. For a regex, use the syntax: `match=<regex>`
IMSI upper bound (input)	The upper bound for the IMSI value	For a numeric range, enter the IMSI value. For a regex, leave it blank.
SRK (output)	The SRK key	Any string value

Examples:

- For configuring numeric range between 9840510345 to 984059999: Lower bound: 9840510345, Upper bound: 9840598823

- For configuring regex for numbers in range 9840500000 to 9840599999: Lower bound: match=98405[0-9]*, Upper bound : <leave it empty>
- For configuring regex for numbers in range 9840501333 to 9840502999: Lower bound: match=984050(1|2)[3-9]* , Upper bound : <leave it empty>
- For configuring regex for numbers in range 9840500000 to 9840599999: Lower bound: match=98405(([2-7][0-9]*)|(8[0-8][0-4][0-5][0-6])|(1[0-9][2-9][3-9][4-9])), Upper bound : <leave it empty>

Figure 18: IMSI Range



MSISDN Range

The MSISDN Range is used in range-based routing to configure the range of MSISDN values.

Table 3: MSISDN Range

Field	Description	Value
Routing Profile (input)	The routing profile name	Any string value
IMSI lower bound (input)	The lower bound for the IMSI value	For a numeric range, enter the IMSI value. For a regex, use the syntax: `match=<regex>`
IMSI upper bound (input)	The upper bound for the IMSI value	For a numeric range, enter the IMSI value. For a regex, leave it blank.
SRK (output)	The SRK key	Any string value

Examples:

- For configuring numeric range between 9840510345 to 984059999: Lower bound: 9840510345, Upper bound: 9840598823
- For configuring regex for numbers in range 9840500000 to 9840599999: Lower bound: match=98405[0-9]*, Upper bound : <leave it empty>
- For configuring regex for numbers in range 9840501333 to 9840502999: Lower bound: match=984050(1|2)[3-9]* , Upper bound : <leave it empty>
- For configuring regex for numbers in range 9840500000 to 9840599999: Lower bound: match=98405(([2-7][0-9]*)|(8[0-8][0-4][0-5][0-6])|(1[0-9][2-9][3-9][4-9])), Upper bound : <leave it empty>

Figure 19: MSISDN Range