Oracle® Communications Diameter Signaling Router

Full Address Based Resolution User Guide

Release 9.0.2.0.0
F92321-01
April 2024

Oracle Communications Diameter Signaling Router Full Address Based Resolution User Guide, Release 9.0.2.0.0

F92321-01

Contents

What's New in this Guide

This section introduces the documentation updates for release 9.0.2.0.0.

Release 9.0.2.0.0 - F92321-01, April 2024

- Updated the connection group `UDRSvcGrp` in UDR Remote Server Configuration.
- Added note about "sip" URI in Request Message Validation.

1
Introduction

The *Full Address Based Resolution (FABR) User's Guide* and Help provides an overview of the functions and procedures to configure FABR. The contents of this chapter include sections on the scope, audience, and organization of the documentation, and how to contact Oracle for assistance.

1.1 Overview

The Full Address Based Resolution (FABR) documentation provides information about functions, and how to use the GUI and the following procedures to configure the application:

- Applications
- Exceptions
- Default Destinations
- Address Resolutions
- System Options

1.2 Scope and Audience

The FABR documentation is intended for anyone responsible for configuring and using the Full Address Based Resolution application. Users of this manual must have a working knowledge of telecommunications, of network installations, and of the product that is using the FABR functions.

1.3 Manual Organization

This manual is organized into the following chapters:

- Introduction contains general information about the FABR help documentation, the organization of this manual, and how to get technical assistance.
- Full Address Based Resolution describes the function of the FABR application.
- Configuration of FABR describes how to configure the FABR application, including Applications, Exceptions, Default Destinations, Address Resolutions, and System Options.
- Maintenance of FABR describes maintenance functions and information that can be used with the FABR application.

1.4 My Oracle Support

My Oracle Support (https://support.oracle.com) is your initial point of contact for all product support and training needs. A representative at Customer Access Support can assist you with My Oracle Support registration.

Call the Customer Access Support main number at 1-800-223-1711 (toll-free in the US), or call the Oracle Support hotline for your local country from the list at http://www.oracle.com/us/support/contact/index.html. When calling, make the selections in the sequence shown below on the Support telephone menu:

1. Select **2** for New Service Request.
2. Select **3** for Hardware, Networking and Solaris Operating System Support.
3. Select one of the following options:
    - For Technical issues such as creating a new Service Request (SR), select **1**.
    - For Non-technical issues such as registration or assistance with My Oracle Support, select **2**.

You are connected to a live agent who can assist you with My Oracle Support registration and opening a support ticket.

My Oracle Support is available 24 hours a day, 7 days a week, 365 days a year.

2
Full Address Based Resolution

This section provides an overview of the functions for the **Full Address Based Resolution** (**FABR**) application.

2.1 Full Address Based Resolution overview

Full Address Based Resolution (FABR) is a routing application that enables network operators to resolve the designated Diameter server (IMS HSS, LTE HSS, MTC HSS, PCRF, OCS, OFCS, AAA) addresses based on Diameter Application ID, Command Code, Routing Entity Type, and Routing Entity Addresses, and then routes the Diameter Request to the resolved destination.

The FABR application validates the ingress Diameter Request message, retrieves the Application ID and Command Code from it, and determines the desired Routing Entity Type to decode from the message based on the configuration. The FABR application extracts the Routing Entity Address from user-configured Attribute-Value Pairs (AVPs) in the ingress message and sends the Routing Entity Address, if extracted successfully, to an off-board DP running the Subscriber Database Server (SDS) for destination address resolution.

The resolved Destination address can be any combination of a Realm and Fully Qualified Domain Name (FQDN); Realm-only, FQDN-only, or Realm and FQDN.

FABR replaces the Destination-Host and/or Destination-Realm AVP in the ingress Request message with the corresponding values of the resolved Destination, and forwards the message to the Diameter Relay Agent for egress routing into the network.

A Routing Entity can be any of the following:

- International Mobile Subscriber Identity (IMSI)
- Mobile Subscriber Integrated Services Digital Network (Number) (MSISDN)
- IP Multimedia Private Identity (IMPI)
- IP Multimedia Public Identity (IMPU)
- External Identifier

FABR Functions

FABR provides the following functions:

- Routing based on IMSI/MSISDN Prefix Lookup performs prefix-based lookups after completion of the full address lookup. The prefix and range based lookup are only performed if the full address lookup does not find a match and can be enabled by the operator for a combination of Application-ID, Command-Code, and Routing Entity Type. Populates the Destination-Host AVP and/or the Destination-Realm AVP based on the resolved destination if a match is found in the prefix database.

  Performs the No Address Match Found routing exception handling procedure if there is not a match in the prefix database.

  The IMSI/MSISDN Prefix and Range lookup can be enabled or disabled on system wide.

- DP Query Bundling enhances the FABR-to-DP interface by supporting the bundling of multiple queries into a single bundled query stack event when enabled. When the DP receives a bundled query, the corresponding bundled response have responses to all the queries that constitute the bundled query.
- Reserved MCC Ranges are reserved for future Mobile Country Code (MCC) ranges and are defined in a system-wide MCC Ranges table. If the MCC digits portion of decoded IMSI digits fall within one of the ranges designated in the MCC Ranges table, the IMSI digits are NOT used for further address resolution. FABR continues decoding the digits using other AVP instances or next Priority AVP (if provisioned), or next Routing Entity (if provisioned).
- Identifying IMSI, MSISDN, and External Identifier address resolution applications like Full Address Based Resolution (FABR) and Range Based Address Resolution (RBAR) need to categorize User Identities (digit strings) decoded from the Diameter Request AVPs as either MSISDN or IMSI to allow looking up the User Identity in the appropriate lookup table.
  If there is no plus sign before the digits, the Routing Entity Type is IMPU, and decoded digits fall within MSISDN and IMSI overlap range. Configured MCC+MNC combinations can be compared to the first 5 or 6 digits of the User Identity. The User Identity is considered as an IMSI and used for IMSI lookup if a match occurs. The User Identity considered as a MSISDN and used for MSISDN lookup if a match does not occur.

  Identifying IMSIs and MSISDNs provides more information about identifying IMSIs and MSISDNs using digit string lengths and MCC+MNC combinations.
- Application Chaining allows FABR and the DM-IWF applications to process the same Diameter Request message by configuring application routing rules.

2.2 Application Chaining

Application Chaining is a method for invoking multiple DSR applications in sequence on the same DSR.

To process a Request for two DSR applications executing in sequence, the Application Route Table execution is:

1. When the Request enters the system at the Application Routing Table (ART).
2. When DSR Application 1 sends the Request back to the Diameter Routing Function at the Application Routing Table (ART).
3. When DSR Application 2 sends the Request back to the Diameter Routing Function at the Application Routing Table (ART).
   At the Application Routing Table (ART) if there is no matching Application Routing Rule for the Request, the Request routes to Peer Route Table for processing.

- Application Route Table (ART)
  Application Route Tables are used for routing Request messages to DSR applications. An ART contains a prioritized list of user-configurable Application Routing Rules. Each Application Routing Rule associates Request message content with a DSR application.

  An ART is searched, when a received Request message from a Peer Node or a DSR application. Searching an ART when a Request message is received from a DSR application allows the operator to route the ingress Diameter transaction to multiple DSR applications in sequence. The operator can create multiple ARTs to assign an ART to a Request message based upon a set of user-defined criteria.

- Application Routing Rules
  An ART consists of a set of prioritized Application Routing Rules that the Diameter Routing Function searches with the content of a Request message, to determine whether to forward the message to a DSR application for processing.

  One ART is searched each time a Request message received from a Peer Node or a DSR application. This method allows forwarding a Diameter transaction to one or more DSR applications for processing.

  However, the Diameter Routing Function does not allow a DSR application to process a Diameter transaction more than once. The Diameter Routing Function internally keeps track of which DSR applications have already processed the message. When the Diameter Routing Function searches an ART and encounters an Application Routing Rule associated with a DSR application that processed the transaction, the Diameter Routing Function bypasses the Application Routing Rule.

The system default ART is not removable using the configuration GUI. The user can create additional ARTs and then define, through configuration, which ART is searched based on ART precedence selection rules.

Each time a Request message is received from a Peer Node or DSR application, the Diameter Routing Function selects an ART to search based on the following ART precedence selection rules (highest to lowest priority):

1. The ART provided by the DSR application if it exists (applies only when the Request message was received from a DSR application)
2. The ART assigned to the ingress Peer Node from which the Request message was received if it exists
3. The ART assigned to the Diameter Application ID in the Request message header if it exists
4. The default ART

The order of DSR applications which can process an ingress Request message is determined by operator configuration of one or more Application Route Tables.

- Each time the Diameter Routing Function receives a Request message from a Peer Node or DSR application, it searches the Application Route Tables to determine where to forward the message.
- The highest priority Application Routing Rule matched defines where to forward the message.
- If no Application Routing Rule match is found, the Diameter Routing Function begins Relay Agent routing to an upstream Peer Node.

When FABR or RBAR and the Diameter-MAP InterWorking Function (DM-IWF) applications run in the same DA-MP, the same Diameter Request message can be processed by both applications.

For a Diameter-to-MAP Request message received from a Diameter Peer that needs to be processed by FABR followed by DM-IWF, two Application Routing Rules are needed; one for routing the message first to FABR and the second to reroute the message to DM-IWF after FABR processing is complete. In this order:

- After the Request is received from the Peer, the Diameter Routing Function searches the Application Routing Rules for the highest priority-matching rule. If this rule contains the FABR application name, the results route the Request to FABR.
- The FABR processes the message and returns it to the Diameter Routing Function.

- The Diameter Routing Function searches the Application Routing Rules for the highest priority matching rule (excluding all rules that would result in routing of the Request to FABR again).This rule contains the DM-IWF application name and the results route the Request to DM-IWF.
- DM-IWF processes the message and sends it to an MD-IWF application (SS7-MP).

For a MAP-to-Diameter Request message received by DM-IWF from an MD-IWF application (SS7-MP), which needs processing by FABR after DM-IWF processing, a single Application Routing Rule is needed for routing the message to FABR after DM-IWF processing is completed. In this order:

- DM-IWF processes the message and sends it to the Diameter Routing Function.
- The Diameter Routing Function searches the Application Routing Rule for the highest priority matching rule (excluding all rules that would cause routing of the Request to DM-IWF again). This rule contains the FABR application name and results in the Request being routed to FABR for processing.
- FABR returns the message to the Diameter Routing Function to complete the routing process.

2.3 Request Message Validation

The derivation of a user identity address from the ingress diameter request message governed by the rules determined by user identity configuration. The configuration defines the supported application IDs, the supported command codes associated with each application ID, the preferred user identity types to search, and the associated AVPs that contain the user identity addresses.

The FABR application processes the diameter request message based on the configuration to extract the user identity addresses.

The diameter request message sent to FABR validates as followed:

- Determine whether the Application ID in the message header is defined in the configuration.
- If the diameter request message receives a valid (configured) Application ID, validate whether the pair (Application ID and command code) in the message is defined in the configuration.
- If the pair is configured, select the highest priority user identity type associated with the pair in the configuration for user identity address searching.
- Search for a valid user identity address from an AVP in the ingress message based on a prioritized set of AVPs assigned to the triplet (Application ID, command code, and then routing entity type).

If a user identity address cannot be found in searching the configured user identity types and AVPs, the `No Valid Routing Entity Address` routing exception handling procedure invokes.

Routing Exception Handling

When an ingress FABR request message cannot be resolved to a destination, (no address matched, no valid digits decoded, or any other error returns), FABR invokes a routing exception handling procedure based on user-defined configuration.

Routing exception handling procedures result in one of the following configured actions:

- Forward the message unchanged
- Forward the message using a user-defined default destination
- Send an Answer response with a user-defined result-code AVP value
- Send an Answer response with user-defined experimental-code AVP values
- Abandon Request

The routing exceptions types support the following:

- Unknown command code
- No valid routing entity addresses were found
- A valid routing entity address did not resolve to a configured address
- Blacklisted subscriber
- DP congestion
- DP errors

Supported AVPs

FABR supports the AVPs associated with the user identity types as defined in Table 2-1.

> **Note:**
>
> There is no support for Service-Information: Subscription-ID-Data (4-Server Private) in FABR and not looked up when retrieving a user identity address.

Table 2-1 FABR Supported AVPs

AVPs	AVP Code	AVP Type	AVP Reference
User-Identity [Public-Identity] [MSISDN]	700	Grouped	Section 6.3.1 of 3GPP 29.329
MSISDN	701	OctetString	Section 6.3.2 of 3GPP 29.329
Public-Identity	601	UTF8String	Section 6.3.2 of 3GPP 29.229
Service-Information [Subscription-ID] […]	873	Grouped	Section 7.2.192 of 3GPP 32.299
Subscription-Id [Subscription-ID-Type [Subscription-ID-Data]	443	Grouped	Section 8.46 of RFC 4006
Subscription-ID-Type	450	Enumerated	Section 8.47 of RFC 4006
Subscription-ID-Data	444	UTF8String	Section 8.47 of RFC 4006
User-Name	1	UTF8String	Section 8.14 of RFC 3588bis

Table 2-1 (Cont.) FABR Supported AVPs

AVPs	AVP Code	AVP Type	AVP Reference
Wildcarded-Public-Identity	634	UTF8String	Section 6.3.35 of 3GPP 29.229
MSISDN	701	OctetString	Section 6.3.2 of 3GPP 29.329
Public Identity	601	UTF8String	Section 6.3.2 of 3GPP 29.229
User-Identifier: [User-Name] [MSISDN]	3102	Grouped	Section 6.4.2 of 3GPP 29.336
User-Name	1	UTF8String	Section 8.14 of RFC 6733
MSISDN	701	OctetString	Section 6.3.2 of 3GPP 29.329
External Identifier		UTF8String	Section 6.3.2 of 3GPP 29.336

Each of the configured user identity types supported in FABR is associated with certain AVPs that contains the user identity type as defined by various diameter application standards. Table 2-2 presents all possible combinations of the user identity types and the associated AVPs.

Table 2-2 Combinations of User Identity Types and Associated AVPs

User Identity Types/AVPs	IMSI	MSISDN	IMPI	IMPU
MSISDN		Applicable		Applicable
User-Identity: MSISDN		Applicable		Applicable
Public-Identity	Applicable	Applicable	Applicable	Applicable
User-Identity: Public-Identity	Applicable	Applicable	Applicable	Applicable
User-Name	Applicable	Applicable	Applicable	Applicable
Subscription-ID-Data (0-E.164)		Applicable		Applicable
Service-Information: Subscription-ID-Data (0-E.164)		Applicable		Applicable
Subscription-ID-Data (1-IMSI)	Applicable		Applicable	
Service-Information: Subscription-ID-Data (1-IMSI)	Applicable		Applicable	

Table 2-2 (Cont.) Combinations of User Identity Types and Associated AVPs

User Identity Types/AVPs	IMSI	MSISDN	IMPI	IMPU
Subscription-ID-Data (2-SIP URI)	Applicable	Applicable	Applicable	Applicable
Service-Information:	Applicable	Applicable	Applicable	Applicable
Subscription-ID-Data (2-SIP URI)				
Subscription-ID-Data (3-NAI)	Applicable	Applicable	Applicable	Applicable
Service-Information:	Applicable	Applicable	Applicable	Applicable
Subscription-ID-Data (3-NAI)				
Wildcarded-Public-Identity				Applicable
User-Identifier: User-Name	Applicable		Applicable	
User-Identifier: MSISDN		Applicable		Applicable
User-Identifier: External Identifier	Applicable	Applicable		

A user identity type can be associated with one or more data formats that is examined when deriving the user identity address from the associated AVPs. The relation between user identity types and the corresponding data formats to be encountered in the ingress diameter request message are listed in Table 2-3.

Table 2-3 Relation between Configured User Identity Types and Data Formats

Configurable User Identity Types/User Identity Formats in Messages	IMSI	MSISDN	IMPI	IMPU
IMSI Format: ASCII Example: 311480123456789	Applicable		Applicable	
MSISDN Format: ASCII and TBCD Example: 19194605500		Applicable		Applicable
SIP URI with IMSI Format: ASCII	Applicable		Applicable	

Table 2-3 (Cont.) Relation between Configured User Identity Types and Data Formats

Configurable User Identity Types/User Identity Formats in Messages	IMSI	MSISDN	IMPI	IMPU
Examples: sip:123456789012345@nai.epc.mnc456.mcc123.3gppnetwork.org sip:6311150999995555@ims.mnc015.mcc311.3gppnetwork.org sip:311480999995555@my.network.org sip:6311480999995555@my.network.org				
SIP URI with MSISDN Format: ASCII Examples: sip:+1-919-460-5500@xyz.com;user=phone sip:311480999995555@my.network.org		Applicable		Applicable
SIP URI with NAI Format: ASCII Example: sip:311480999995555@mynetwork.org			Applicable	Applicable
SIP URI with Wildcarded PSI Format: ASCII Example: sip:WP-A_ServiceType-!.*!@att.com				Applicable
TEL URI with MSISDN FORMAT: ASCII Examples: tel:+1-919-460-5500; phone-context=example.com tel:+19258889999 tel:19195551212		Applicable		Applicable
NAI with IMSI/ MSISDN Format: ASCII Examples: 123456789012345@xyz.com 123456789012345 311480999995555@ims.mnc480.mcc311.3gppnetwork.org 6311150999995555@xyz.com 6311150999995555@ims.mnc015.mcc311.3gppnetwork.org	Applicable	Applicable	Applicable	Applicable

Table 2-3 (Cont.) Relation between Configured User Identity Types and Data Formats

Configurable User Identity Types/User Identity Formats in Messages	IMSI	MSISDN	IMPI	IMPU
NAI Format: ASCII Example: handy.manny@xyz.com			Applicable	Applicable

> **Note:**
>
> The "sip:" in sip URI must be in lower case.

Identifying IMSIs and MSISDNs

In certain Diameter messages over the Cx interface (and possibly over the Sh interface), certain AVPs that typically carry an IMSI sometimes can carry an MSISDN.

Address resolution applications like Full Address Based Resolution (FABR) and Range Based Address Resolution (RBAR) need to categorize user Identities (digit strings) decoded from the diameter request AVPs as either MSISDN or IMSI, to allow looking up the user identity in the appropriate lookup table.

Most of the time, these applications can clearly categorize the decoded user identity based on:

- The configured routing entity type
- The contents of the AVP
  For instance, if the user identity has been decoded from a SIP URI that has a plus sign before the digits (such as sig:+1-919-460-5500@oracle.com), it can be directly categorized as an MSISDN.
- The number of digits in the user identity

In certain cases, none of these methods allow a clear categorization (for example, if the number of digits needs to be used and the received number of digits are applicable to both IMSIs and MSISDNs, and thus leads to an ambiguous determination; or if there is no plus sign before the digits).

If FABR has been configured to decode an IMPU from a user identity (digit string), but cannot determine whether the user identity is an IMSI or an MSISDN based on digit analysis, a tie-breaker is needed to properly categorize the user identity.

If the routing entity type is IMPU, the user identity extracted results in only digits, and the length of the digits in the user identity falls within an overlap digits range of MSISDN and IMSI, the following logic can be used to determine if the user identity is an IMSI or MSISDN.

- FABR extracts the first 5 or 6 digits of the user identity and compares them against a list of configured 5- or 6-digit MCC-MNC combinations.

The **Diameter Common**, and then **Network Identifiers**, and then **MCCMNC** pages can be used to configure up to 2500 distinct combinations of Mobile Country Code (MCC) and Mobile Network Code (MNC). (Refer to *Diameter Common User's Guide* and Help for procedures to configure MCC-MNC combinations.)

- If a match occurs, the user identity is considered as an IMSI and used for IMSI lookup.
- If a match does not occur, the user identity is considered as an MSISDN and used for MSISDN lookup.

2.4 Multiple DSR Application Invocation Prevention

The DSR provides a mechanism for preventing the same DSR application from being invoked on two different DSR nodes:

- When a DSR application does not want to be invoked a second time on another DSR, it can insert a DSR AVP called DSR-Application-Invoked containing its DSR Application ID.
- When the Diameter Routing Function searches an ART, it ignores any Application Routing Rules associated with a DSR application that has inserted the DSR-Application-Invoked AVP

DSR-Application-Invoked AVP

To prevent the same DSR application from being invoked on multiple DSRs in a network (and processing the same message twice by the same DSR application), a DSR application can (optionally) add to the Request message a DSR-Application-Invoked AVP containing the DSR Application ID as shown in DSR Application-Invoked AVP.

Table 2-4 DSR Application-Invoked AVP

Byte 1	Byte 2	Byte 3	Byte 4
AVP Code = 2468			
Flags=10000000		Length = 16	
Vendor ID = 323			
DSR Application ID = Unsigned32			

This AVP is decoded by the Diameter Routing Function before ART processing to prevent multiple invocations of the same DSR application. Any Application Routing Rule with this DSR Application ID is ignored by the Diameter Routing Function.

This AVP can be repeated in the Request to indicate different DSR applications, but is inserted only once per DSR application.

Insertion of a DSR Application-Invoked AVP is controlled by DSR application specific configuration on the **FABR**, and then **Configuration**, and then **System Options** GUI page, such as:

Allow Subsequent FABR Invocation - Checked = Yes, Unchecked = No

If checked, subsequent invocation of FABR on a different DSR node in the network is allowed.

2.5 Transaction Metadata Recording for Integrated DIH (IDIH)

Integrated Diameter Intelligence Hub (**IDIH**) can be used to capture detailed information about selected Diameter transactions, and transmit this information to **DIH** for further analysis.

The Diameter Routing Function and invoked DSR applications record detailed information about each Diameter transaction - called transaction metadata. Each metadata record describes an important event in the lifetime of a Diameter transaction. Metadata appears in the Trace Transaction Record (**TTR**) in the order that the metadata-generating events actually occurred. Together, all of the metadata records combine to document the processing performed on the entire transaction, and can later be used to provide diagnostic information when performing troubleshooting. Metadata is recorded to a TTR for each transaction so that, even if the transaction is selected to be sent to DIH at an Answer Troubleshooting Trigger Point (TTP-IA or TTP-EA), the metadata for all of the messages in the transaction is present.

The functions of IDIH are described in the *Integrated DIH User's Guide* and Help.

FABR records the application-specific metadata events described in Table 2-5.

Table 2-5 FABR Metadata-Generating Events

Event	Type	Scope	Instance Data	When Recorded
Address Resolution Match found	Address Resolution Match	App Data	• Routing Entity Type (such as IMSI) • Routing Entity AVP (such as Public-Identity) • Routing Entity Address (such as 311480123456789)	After FABR searches and finds a valid Routing Entity address in an ingress Request message using a prioritized set of AVPs associated with the highest priority Routing Entity Type assigned to the Address Resolution order pair (Diameter Application ID, Command Code).
DP Query Event Sent to DP for processing	DP Query Sent	App Data	• Routing Entity Data Format (such as IMSI) • Routing Entity Address (such as 123456789012345) • Destination Type (such as IMS-HSS)	When FABR sends a DP query event to the DP for Destination address resolution.
DP Response Event Received from DP	DP Response Received	App Data	• DP IP Address Type (such as IPv4) • DP IP Address (such as 10.240.55.25) • Result Code String (such as Blacklisted) • Destination Realm (such as xyz.com) • Destination FQDN (such as hss1.hss.xyz.com)	When FABR receives a response to a previous DP query.

Table 2-5 (Cont.) FABR Metadata-Generating Events

Event	Type	Scope	Instance Data	When Recorded
Routing Exception	Routing Exception	App Data	• Routing Exception Type (such as DP Congestion) • Routing Exception Action (such as Abandon Request)	After any Routing Exception is encountered.
DP Query Failure	DP Query Failure	App Data	• DP IP Address Type (such as IPv4) • DP IP Address (such as 10.240.55.25)	After any DP Query failure other than a response timeout is encountered.
DP Response Timed out	DP Response Timeout	App Data	• DP IP Address Type (such as IPv4) • DP IP Address (such as 10.240.55.25)	When FABR times out waiting to receive a response from the DP to a previous Destination address resolution query.

2.6 FABR with User Data Repository

FABR application supports sending ComAgent stack events to User Data Repository (UDR) NOAM. FABR with UDR supports bundling and non-bundling cases. DSR can opt for SDS or UDR as database type for FABR queries.

Limitations

As DSR can now send FABR queries to UDR, unlike SDS the following features are currently not supported by UDR:

- Prefix Search for IMSI and MSISDN
- External Identifier match partial (Domain identifier)
- 16 priority Support
- NGN PS priority Support
- Wild Card Nai User

3
Configuration of FABR

This section describes the procedures used to configure the FABR application.

3.1 Pre-Configuration Activities

Before FABR configuration can be performed, the following activities need to be performed in the system:

- Verify server status
- Gather information required for Diameter, Diameter Common, and FABR configuration
- Configure Diameter Common components required for FABR configuration
- Configure Diameter components required for FABR configuration
- Configure SDS DP / UDR NOAM Remote servers in Communication Agent (**ComAgent**)

3.1.1 Verifying Server status

Use this task to verify server status before FABR configuration.

1. From the active **SOAM** in a DSR topology, click **Status & Manage**, and then **Server**.
2. Verify that for each server the **Appl State** field is **Disabled**, and the **DB**, **Reporting Status**, and **Proc** fields are **Norm**.

3.1.2 Diameter Common Configuration for FABR

The following Diameter Common configuration must be done before FABR configuration can be performed.

Use the *Diameter Common User's Guide* to complete the Diameter Common configuration, including the Diameter Common components needed for use with FABR.

SOAM Diameter Common Configuration

Diameter Common configuration for MCC Ranges Network Identifiers and MP Profile assignment for FABR is done from the **SOAM GUI** in a DSR topology.

1. **MPs**
   Click **Diameter Common**, and then **MPs**, and then **Profile Assignments**, and verify the correct Database MP Profiles have been assigned for FABR DA-MPs shown in the DA-MP list. If assignments need to be made or changed, use the **Diameter Common**, and then **MPs**, and then **Profile Assignments** page to assign the correct MP Profiles.
2. **MCC Ranges**
   Use the **Diameter Common**, and then **Network Identifiers**, and then **MCC Ranges [Insert]** page to specify up to 10 distinct, non-overlapping MCC Ranges.

   The following two MCC Ranges are reserved by telephony standards and are recommended to be created in addition to other specified ranges:

a. 000-199
b. 800-899

NOAM Diameter Common Configuration

Diameter Common configuration for MCCMNC Network Identifiers for FABR is done from the **NOAM** GUI in a DSR topology.

1. Use the **Diameter Common**, and then **Network Identifiers**, and then **MCCMNC [Insert]** page to configure MCCMNC entries.

3.1.3 Diameter Configuration for FABR

The following Diameter configuration must be done before FABR configuration can be performed.

All Diameter Configuration is done using the **SOAM GUI** in a DSR topology.

Use the *Diameter User's Guide* to complete the Diameter configuration, including the Diameter components needed for use with FABR.

1. **Application IDs**
   Diameter Application IDs must be configured prior to making them available for use in a FABR Address Resolution. Use the **Diameter**, and then **Configuration**, and then **Application Ids [Insert]** page to configure Diameter Application IDs.

   The Application IDs that need to be configured depend on the types of Diameter servers being supported, including **HSS**, **PCRF**, **OFCS**, **OCS**, and **AAA**.

2. **Command Codes**
   Diameter Command Codes must be configured prior to using them in a FABR Address Resolution. Use the **Diameter**, and then **Configuration**, and then **Command Codes [Insert]** page to configure Diameter Command Codes.

   Configure any Command Codes that need to be handled by FABR. The Command Codes are associated with the Diameter applications supported by the Diameter servers (for example, HSS, PCRF, OCFS, OCS, or AAA), which are the destination of Diameter Requests being routed by FABR. For example, the combination of Application ID = S6a and Command Code = ULR/ULA might be relevant for HSS.

3. **Application Route Tables**
   Either use the default Application Route Table (always available), or use the **Diameter**, and then **Configuration**, and then **Command Codes**, and then **Application Route Tables [Insert]** page to configure one or more Application Route Tables in addition to the default. Application Route Tables contain Application Routing Rules that direct messages to FABR and other DSR applications.

4. **Application Routing Rules**
   On the **Diameter**, and then **Configuration**, and then **Command Codes**, and then **Application Route Tables** page, select an Application Route Table Name and click **View/Edit Rules**.

   Use the Viewing Rules for Application Route Table page to insert or edit an Application Routing Rule with the Application Name set to FABR so messages are directed to FABR.

If the FABR application and the DM-IWF application are chained so both of them can process the same Request message, insert or edit a second Application Routing Rule with the Application Name set to DM-IWF.

Set the Priority in each of the two Application Routing Rules to indicate which application processes the message first (the higher priority processes first).

- Set the **Application Name** to **FABR**.
- In the **Conditions** field, set the **Application-Id Operator** to**Equals** and the**Value** to **4**. For all other parameters, set the**Operator** to **Always True**.

3.1.4 SDS DP Remote Server Configuration

Use this procedure to configure **SDS DP** Remote Servers to allow FABR to use SDS for address lookup and resolution.

1. From the active **NOAM**, click **Communication Agent**, and then **Configuration**, and then **Remote Servers**.
2. Click **Insert**.
3. Enter a unique **Remote Server Name**.
4. Enter the **Remote Server IP Address**.

   Specify the IP address that can be reached via a server's Internal Management Interface (IMI). The IP address uniquely identifies the Remote Server and provides the means by which **Communication Agent** can establish transport connections to/from the Remote Server.
5. For **Remote Server Mode**, select **Server**.
6. Assign the Remote Server to one of the **Available Local Server Groups**.
7. Click **Ok**.
8. Select **Communication Agent**, and then **Configuration**, and then **Connection Groups**
9. Select the `DPSvcGroup` and click **Edit**.
10. Assign the Remote Server you just created to the DPSvcGroup Connection group.
11. Click **Ok**.
12. Expand the **Servers** assigned to the `DPSvcGroup` to see that the new Remote Server is now included.

The operational status of what was provisioned can be verified by using the **Communication Agent**, and then **Maintenance** pages.

- Click **Communication Agent**, and then **Maintenance**, and then **Connection Status** to verify all Remote Server connections added are shown as `InService` on all local servers.
- Click **Communication Agent**, and then **Maintenance**, and then **Routed Service Status** to verify the status is `Available` for all local servers that are provisioned to connect.

3.1.5 UDR Remote Server Configuration

Use this procedure to configure **UDR** Remote Servers to allow FABR to use UDR for address lookup and resolution.

1. From the active **NOAM**, click **Communication Agent**, and then **Configuration**, and then **Remote Servers**.
2. Click **Insert**.
3. Enter a unique **Remote Server Name**.
4. Enter the **Remote Server IP Address**.

   Specify the IP address that can be reached via a server's Internal Management Interface (IMI). The IP address uniquely identifies the Remote Server and provides the means by which **Communication Agent** can establish transport connections to/from the Remote Server.
5. For **Remote Server Mode**, select **Server**.
6. Assign the Remote Server to one of the **Available Local Server Groups**.
7. Click **Ok**.
8. Select **Communication Agent**, click **Configuration** and then select **Connection Groups.**
9. Select the `UDRSvcGrp` and click **Edit**.
10. Assign the Remote Server you just created to the `UDRSvcGrp` Connection group.
11. Click **Ok**.
12. Expand the **Servers** assigned to the `UDRSvcGrp` to see that the new Remote Server is now included.
13. Restart the Message Processor.

The operational status of what was provisioned can be verified by using the **Communication Agent**, and then **Maintenance** pages.

- Click **Communication Agent**, and then **Maintenance**, and then **Connection Status** to verify all Remote Server connections added are shown as `InService` on all local servers.
- Click **Communication Agent**, and then **Maintenance**, and then **Routed Service Status** to verify the status is `Available` for all local servers that are provisioned to connect.

3.2 FABR Configuration

The **FABR**, and then **Configuration** pages allow you to manage FABR application configuration.

FABR configuration typically occurs in the following order:

1. Add Diameter **Applications** to a list of FABR supported Diameter applications.
2. If necessary, configure **Default Destinations**.
3. If necessary, edit routing **Exceptions**.

> **Note:**
>
> If a **Routing Exception Action** of **Forward Unchanged** is configured, configure a **Default Destination**.

4. Configure **Address Resolutions**.
5. If necessary, change the **System Options**.

3.2.1 Applications configuration

The **Applications** page under **Configuration** for **FABR** allows you to access a list of Diameter applications supported by FABR.

From the FABR > Configuration > Applications page, you can:

- Filter the list of supported Diameter applications to display only the desired application(s).
- View a list of supported Diameter applications.
- Insert a supported Diameter application.

> **Note:**
>
> When an Application entry is added, Routing Exceptions (Unknown Command Code, No valid Routing Entity Address, No Address Match) are automatically inserted with the Routing Exception Action value as Forward Unchanged.

- Delete a Diameter application from the list of supported Diameter applications.

> **Note:**
>
> When an Application entry is deleted, the associated Routing Exceptions are automatically deleted.

3.2.1.1 Applications configuration elements

Table 3-1 describe the fields on the Applications insert page.

Table 3-1 Applications Configuration Elements

Field	Description	Data Input Notes
*Application ID	Diameter Application ID, command code, and routing entity type are useful to determine address resolution for routing request messages.	Format: list Range: Configured Diameter Application IDs Default: none
*Routing Mode (Read only)	Method of routing for request messages received that contain the diameter Application ID.	Format: Disabled list with a value of **Proxy**.

3.2.1.2 Inserting a supported Diameter application

Use this task to add a new Diameter application.

Inserting a supported Application automatically adds Routing Exceptions (**Unknown Command Code**, **No valid Routing Entity Address**, **No Address Match Found**, **DP**

Errors, and **DP Congestion**) with the **Routing Exception Action** set to `Forward Unchanged`.

1. Click **FABR**, and then **Configuration**, and then **Applications**.
2. Click **Insert**.
3. Select **Application ID** from the list.

> **Note:**
>
> The Application IDs presented in this list are those created using **Diameter**, and then **Configuration**, and then **Application IDs**.
>
> Note the **Routing Mode** field is disabled.

4. Click **OK**, **Apply**, or **Cancel**.

3.2.1.3 Deleting a Diameter application from the list of supported Diameter applications

Use this task to delete a Diameter application from the list of supported Diameter applications.

An application cannot be deleted if it is being used by an Address Resolution. Before you perform this task, delete any Address Resolution that uses the Application.

1. Select **FABR**, and then **Configuration**, and then **Applications**.
2. Select the application you want to delete and click **Delete**.

> **Note:**
>
> An error message appears if the application has already been removed.

3. Click **OK** or **Cancel** on the confirmation screen.

3.2.2 Exceptions configuration

The **FABR**, and then **Configuration**, and then **Exceptions** page allows you to specify the routing procedure to invoke when FABR is unable to resolve an address to a Destination for each supported Diameter application and Routing Exception Type.

There are Routing Exception entries automatically inserted with the **Routing Exception Action** set to `Forward Unchanged` as the default action for a supported Diameter application entry when that application entry is added.

- **Unknown Command Code**
- **No valid Routing Entity Address**
- **No Address Match Found**
- **DP Errors**
- **DP Congestion**

- **Blacklist**

Similarly, these Routing Exceptions that are associated with an application entry are automatically deleted when that application entry is deleted.

From the **FABR**, and then **Configuration**, and then **Exceptions** page, you can:

- Filter the list of exceptions to display only the desired exceptions.
- View a list of supported Diameter applications and their associated Routing Exception Types and Routing Exception Actions.
- Edit the Routing Exception Action and its associated attributes for a supported Diameter application.

3.2.2.1 Exceptions configuration elements

Table 3-2 describes the fields on the Exceptions edit page.

Table 3-2 Exceptions Configuration Elements

Field	Description	Data Input Notes
*Application ID	Application ID in a diameter message - Read only	none
Application Name	Name of the application corresponding to the Application ID - Read only.	none
*Routing Exception Type	The routing exception that prevented address resolution - Read only. This field displays one of the following values: • Unknown Command Code • No Valid Routing Entity Address • No Address Match Found • DP Errors • DP Congestion • **Blacklist**ed Subscriber	none
Routing Exception Action	Action that FABR takes associated with the **Routing Exception Type**	Format: options Range: • Forward Unchanged • Forward to Destination • Send Answer with Result-Code AVP • Send Answer with Experimental-Result AVP • Abandon Request
Destination	Destination to where the message is forwarded associated with the **Routing Exception Type**. This field is enabled when the **Routing Exception Action** is set to `Forward to Destination`.	Format: list Range: Available user-configured destinations

Table 3-2 (Cont.) Exceptions Configuration Elements

Field	Description	Data Input Notes
Result-Code Value	Result code associated with this **Routing Exception Type**. This field is enabled when the **Routing Exception Action** is set to either `Send Answer with Result-Code AVP` or `Send Answer with Experimental-Result AVP`.	Format: • field • list Range: • field: 1000–5999 • list with available diameter result codes Default: none
Vendor-ID	Value returned in the Vendor-ID AVP of the answer message associated with this **Routing Exception Type**. This field is enabled when the **Routing Exception Action** is set to `Send Answer with Experimental-Result AVP`.	Format: field Range: 1–4294967295 Default: none
Error Message	Value returned in the Error-Message AVP of the answer message. This field is enabled when the **Routing Exception Action** is set to either `Send Answer with Result-Code AVP` or `Send Answer with Experimental-Result AVP`.	Range: 0–64 characters Default: null string, no Error-Message AVP in Answer message

3.2.2.2 Editing a Routing Exception

Use this task to edit a Routing Exception.

1. Click **FABR**, and then **Configuration**, and then **Exceptions**.
2. Select the Application ID/Name you want to edit and click **Edit**.

> **Note:**
>
> An error message appears if the application has already been removed.

3. Update the relevant fields.

   For more information about each field, see Table 3-2.

   - An error is displayed if **Vendor-ID** is not configured when `Send Answer with Experimental-Result` AVP is selected as a value for **Routing Exception Action**.
   - An error is displayed if **Destination** is not configured when `Forward to Destination` is selected as a value for **Routing Exception Action**.
   - An error is displayed if **Result-Code Value** is not configured when `Send Answer with Result-Code` AVP or `Send Answer with Experimental-Result` AVP is selected as a value for **Routing Exception Action**.

4. Click **OK**, **Apply** or **Cancel**.

3.2.3 Default Destinations configuration

The FABR > Configuration > Default Destinations page contains the attributes associated with a Default Destination to which FABR routes a message. FABR uses these attributes to modify the contents of a received message before forwarding the message.

Each Default Destination can be configured with any combination of a Realm and FQDN such as Realm-only, FQDN-only, or Realm and FQDN.

From the FABR > Configuration > Default Destinations page, you can:

- Filter the list of Default Destinations to display only the desired destinations.
- View a list of Default Destinations.
- Insert a Default Destination.
- Edit a Default Destination.
- Delete a Default Destination.

3.2.3.1 Default Destinations configuration elements

Table 3-3 describes the fields on the Default Destinations insert and edit pages.

Table 3-3 Destinations Configuration Elements

Field	Description	Data Input Notes
***Name**	Unique name of the destination	Format: field Range: 1–32 characters
Realm	**Realm** of the default destination The realm and fully qualified domain name cannot both be empty; otherwise, an error message appears.	Format: field Range: A valid Realm or FDQN. Default: none
Fully Qualified Domain Name	Unique fully qualified domain name of the default destination If a duplicate FQDN is entered, an error message appears. The **Fully Qualified Domain Name** and **Realm** cannot both be empty; otherwise, an error message appears.	

3.2.3.2 Inserting a Default Destination

Use this task to add a new Default Destination.

1. Click **FABR**, and then **Configuration**, and then **Default Destinations**.
2. Click **Insert**.
3. Enter a unique name for the destination in the **Name** field.
4. Enter the realm in the **Realm** field.
5. Enter a unique FQDN in the **Fully Qualified Domain Name** field.
6. Click **OK**, **Apply** or **Cancel**.

3.2.3.3 Editing a Default Destination

Use this task to edit a Default Destination.

1. Click **FABR**, and then **Configuration**, and then **Default Destinations**.
2. Select the Destination you want to edit and click **Edit**.

> **Note:**
>
> An error message appears if the Destination has already been removed.

3. Update the relevant fields.

   For more information about each field, see Table 3-3.
   The **Name** field is read-only and cannot be edited.
4. Click **OK**, **Apply** or **Cancel**.

3.2.3.4 Deleting a Default Destination

Use this task to delete a Default Destination. A Default Destination cannot be deleted if it is being used by a Routing Exception. Before this task is performed, delete the association with any Routing Exception either by changing the Routing Exception Action to something other than `Forward To Destination`, or by deleting the supported application, thereby deleting the associated Routing Exceptions.

1. Click **FABR**, and then **Configuration**, and then **Default Destinations**.
2. Select the Default Destination you want to delete and click **Delete**.
3. Click **OK** or **Cancel** on the confirmation screen.

3.2.4 Address Resolutions configuration

FABR performs off-board database lookups for user identities decoded from Diameter messages. The **FABR**, and then **Configuration**, and then **Address Resolutions** page allows you to configure which (and how) user identities are to be decoded from the messages. You can provision combinations of Diameter Application ID, and Command Code (the key matched to the messages) and configure the Routing Entity Type(s) to be decoded and a prioritized list of AVPs from which to decode these entity types. An Address Resolution supports up to three prioritized Routing Entity Types for each Application ID and Command Code (from highest priority to lowest priority - Primary Routing Entity Type, Secondary Routing Entity Type, and Tertiary Routing Entity Type) .

From the **FABR**, and then **Configuration**, and then **Address Resolutions** page, you can:

- Filter the list of address resolutions to display only the desired records.
- View a list of address resolutions.
- Insert an address resolution.
- Edit an address resolution.
- Delete an address resolution.

3.2.4.1 Address Resolutions configuration elements

Table 3-4 describes the fields on the Address Resolutions insert and edit pages. Data input notes only apply to the insert and edit pages.

Table 3-4 Address Resolutions Configuration Elements

Field	Description	Data Input Notes
*Application ID	Application ID in a diameter message. The application ID is an **IANA**-assigned diameter application ID, which is a 32-bit field that is mandatory in all diameter messages. It is commonly used for screening and routing messages between diameter nodes. If a combination of the Application ID and command code already exists, an error message appears.	Format: list Range: application IDs configured for FABR
*Command Code	Command Code in a diameter message If a combination of the Application ID and command code already exists, an error message appears.	Format: list Range: command codes configured for diameter
Primary Routing Entity, **Secondary Routing Entity**, **Tertiary Routing Entity** sections		
*Routing Entity	Routing entity type. The same routing entity type cannot be selected for both the primary and the secondary routing entity; if the same type is selected, an error message appears. If the routing entity type is not specified for the primary routing entity, an error message appears.	Format: list Range: • IMSI • MSISDN • IMPI • IMPU • External Identifier
*Primary AVP	Primary AVP used for extracting the routing entity address. The same primary AVP and secondary AVP cannot be selected for either the primary routing entity or for the secondary routing entity; if the same AVP is selected, an error message appears. If primary AVP is not selected for the primary routing entity, an error message appears.	Format: list Range: • Public Identity • ServiceInfo.Subscription-ID(0) • ServiceInfo.Subscription-ID(1) • ServiceInfo.Subscription-ID(2) • ServiceInfo.Subscription-ID(3) • Subscription-ID(0) • Subscription-ID(1) • Subscription-ID(2) • Subscription-ID(3) • UserIdentifier.External-Identifier • UserIdentifier.MSISDN • UserIdentity.MSISDN • UserIdentity.Public-Identity • MSISDN • UserName • UserIdentifier.External-Identifier

Table 3-4 (Cont.) Address Resolutions Configuration Elements

Field	Description	Data Input Notes
Secondary AVP	Secondary AVP used for extracting the routing entity address. The same primary AVP and secondary AVP cannot be selected for either the primary routing entity or for the secondary routing entity; if the same AVP is selected, an error message appears.	• Wildcarded-Public-Identity
*Destination Type	Type of Destination for this routing entity type.	Format: list Range: • AAA • IMS-HSS • LTE-HSS • MTC-HSS • OCS • OFCS • PCRF • USERDEF1 • USERDEF2
Prefix Search Enabled	Enables the IMSI/MSISDN prefix based lookup to be performed if the full address lookup did not find a match.	Format: check box Default: none
Blacklist Search Enabled	Enables the IMSI/MSISDN blacklist lookup to be performed before the full address lookup.	Format: check box Default: none
FallBack Search Enable	Enables the Domain-Identifier lookup to be performed when the external identifier lookup did not find an exact match.	Format: check box Default: none

3.2.4.2 Inserting an Address Resolution

Use this task to add a new Address Resolution.
Before this task is performed, make sure there is at least one supported Diameter application configured in the system.

1. Click **FABR**, and then **Configuration**, and then **Address Resolutions**.
2. Click **Insert**.
3. Select the **Application ID** from the list.

> **Note:**
>
> The Application IDs presented in this list are those created using **FABR**, and then **Configuration**, and then **Applications**.

4. Select the **Command Code** from the list.

> **Note:**
>
> The Command Codes presented in this list are those created using **Diameter**, and then **Command Codes**.

5. For the Primary Routing Entity section, perform the following:
   a. Select the **Routing Entity** from the list.
   b. Select the **Primary AVP** from the list.
   c. If needed, select the **Secondary AVP** from the list.
   d. Select the type of destination from the **Destination Type** list.
   e. Check the **Prefix Search Enabled** checkbox to perform the IMSI/MSISDN prefix based lookup.
   f. Check the **Blacklist Search Enabled** checkbox to perform the IMSI/MSISDN blacklist lookup.
   g. Check the **FallBack Search Enabled** checkbox to perform the Domain-Identifier lookup.
6. If needed, for the Secondary Routing Entity section, perform the following:
   a. Select the appropriate Routing Entity type from the **Routing Entity Type** list.
   b. Select the Primary AVP from the **Primary AVP** list.
   c. If needed, select the Secondary AVP from the **Secondary AVP** list.
   d. Select the type of destination from the **Destination Type** list.
   e. Check the **Prefix Search Enabled** checkbox to perform the IMSI/MSISDN prefix based lookup.
   f. Check the **Blacklist Search Enabled** checkbox to perform the IMSI/MSISDN blacklist lookup.
   g. Check the **FallBack Search Enabled** checkbox to perform the Domain-Identifier lookup.
7. If needed, for the Tertiary Routing Entity section, perform the following:
   a. Select the appropriate Routing Entity type from the **Routing Entity Type** list.
   b. Select the Primary AVP from the **Primary AVP** list.
   c. If needed, select the Secondary AVP from the **Secondary AVP** list.
   d. Select the type of destination from the **Destination Type** list.
   e. Check the **Prefix Search Enabled** checkbox to perform the IMSI/MSISDN prefix based lookup.
   f. Check the **Blacklist Search Enabled** checkbox to perform the IMSI/MSISDN blacklist lookup.
   g. Check the **FallBack Search Enabled** checkbox to perform the Domain-Identifier lookup.
8. Click **OK**, **Apply**, or **Cancel**.

3.2.4.3 Editing an Address Resolution

Use this task to edit an Address Resolution.

1. Click **FABR**, and then **Configuration**, and then **Address Resolution**.
2. Select the Address Resolution you want to edit and click **Edit**.

> **Note:**
>
> An error message appears if the Address Resolution has already been removed.

3. Update the relevant fields.

   For more information about each field, see Table 3-4.
   The following fields are read-only and cannot be edited:

   - **Application ID**
   - **Command Code**

4. Click **OK**, **Apply**, or **Cancel**.

3.2.4.4 Deleting an Address Resolution

Use this task to delete an Address Resolution.

1. Click **FABR**, and then **Configuration**, and then **Address Resolutions**.
2. Select the Address Resolution you want to delete and click **Delete**.
3. Click **OK** or **Cancel** on the confirmation screen.

3.2.5 System Options configuration

The System Options page allows you to modify the default system values for FABR global parameters, for example, FQDN/Realm, Allow Subsequent FABR Invocation, or Application Unavailable action.

3.2.5.1 System Options elements

Table 3-5 describes the fields on the System Options page.

Table 3-5 System Options Elements

Field	Description	Data Input Notes
ASCII Excluded Digits	List of **ASCII** characters to ignore while parsing MSISDN digits from a raw AVP data field of AVP Type UTF8String. If an invalid character is entered, an error message appears.	Format: field Default: none Range: ASCII printable characters except percentage sign (%), commercial sign (@), colon sign (:), and semi-colon (;).

Table 3-5 (Cont.) System Options Elements

Field	Description	Data Input Notes
Exclude Space	Defines whether ASCII character space is ignored while parsing MSISDN digits from a raw AVP data field of AVP Type UTF8String If checked, ASCII character space is ignored. If not checked, ASCII character space is not ignored.	Format: check box Default: unchecked
TBCD Excluded Digits	Defines whether the associated digits is ignored while parsing digits from a raw AVP data field of AVP Type OctetString encoded as a **TBCD**-string If checked, digits are ignored. If not checked, digits are not ignored.	Format: check boxes Range: checked, unchecked for each option: *(1010), #(1011), a(1100), b(1101), c(1110) Default: all unchecked
Allow Subsequent FABR Invocation	Enables the subsequent invocation of FABR on a different **DSR** node in the network.	Format: check box Default: unchecked
Remove Destination-Host	If checked, FABR deletes any instance of `Destination-Host` AVPs in the message when performing Realm only resolution.	Format: check box Default: unchecked
Realm	Value to be placed in the Origin-Realm AVP of the Answer message generated by FABR. A Realm must be paired with a Fully Qualified Domain Name. If entering a value for Realm, then a value for Fully Qualified Domain Name must also be entered; otherwise, an error message appears. If a value is not entered, the local node Realm for the egress connection is used.	Format: field Range: A valid Realm Default: none
Fully Qualified Domain Name	Value to be placed in the Origin-Host AVP of the Answer message generated by FABR. A Fully Qualified Domain Name must be paired with a Realm. If entering a value for Fully Qualified Domain Name, then a value for Realm must also be entered; otherwise, an error message appears. If not configured, local node FQDN for the egress connection is used.	Format: field Range: A valid FQDN - up to 255 characters; label-up to 63 characters. Default: none
Resource Exhaustion Result-Code	Result-Code or Experimental-Result-Code value to be returned in an Answer message when a message is not successfully routed because of internal resource being exhausted. If Vendor-ID is configured, this result-code value is encoded as Experimental-Result-Code AVP; otherwise the result-code is encoded as Result-Code AVP.	Format: • field • list Range: • field: 1000–5999 • list with available Code values Default: 3004
Resource Exhaustion Error Message	Error-Message AVP value to be returned in an Answer message when a message is not successfully routed because of internal resource being exhausted.	Range: 0–64 characters Default: FABR Resource Exhausted

Table 3-5 (Cont.) System Options Elements

Field	Description	Data Input Notes
Resource Exhaustion Vendor-ID	Vendor-ID AVP value to be returned in an Answer message when a message is not successfully routed because of internal resource being exhausted.	Format: field Range: 1–4294967295
Application Unavailable Action	Defines action to be taken when FABR is not available to process messages. If the `Default Route` option is selected, an entry must be provided for the Application Unavailable Route List.	Format: options Range: • Continue Routing • Default Route • Send Answer with Result-Code AVP • Send Answer with Experimental-Result AVP Default: Continue Routing
Application Unavailable Route List	Defines where the requests are routed when FABR is not available. Peer Routing Rules are bypassed. A route list must be entered if `Default Route` is selected as the **Application Unavailable Action**.	Format: list Range: Available Route List entries
Application Unavailable Result-Code	Result-Code or Experimental-Result-Code value to be returned in an Answer message when a message is not successfully routed because FABR is not available. If Vendor-ID is configured, this result-code value is encoded as Experimental-Result-Code AVP; otherwise the result-code is encoded as Result-Code AVP. A code must be entered if either the `Send Answer with Result-Code AVP` or the `Send Answer with Experimental Result-Code AVP` option is selected as the **Application Unavailable Action**.	Format: • field • list Range: • field: 1000–5999 • list with available Code values Default: 3002
Application Unavailable Error Message	Error-Message AVP value to be returned in an Answer message when a message is not successfully routed because FABR is not available. A message can be entered, if needed, when either the `Send Answer with Result-Code AVP` or the `Send Answer with Experimental Result-Code AVP` option is selected as the **Application Unavailable Action**.	Range: 0–64 characters Default: FABR Unavailable

Table 3-5 (Cont.) System Options Elements

Field	Description	Data Input Notes
Application Unavailable Vendor-ID	Vendor-ID AVP value to be returned in an Answer message when a message is not successfully routed because FABR is not available. A vendor-ID must be entered if the `Send Answer with Experimental Result-Code AVP` option is selected as the **Application Unavailable Action**.	Format: field Range: 1–4294967295
Bundling Enabled	If enabled, allows FABR to bundle DP query Events to form a DP Bundled query Event to send to DP server.	Format: check box
Maximum Bundle Size	Maximum number of individual DP query Events that can be bundled.	Format: field Range: 2-20 Default: 5
Prefix Search Enabled	If enabled, IMSI/MSISDN prefix based lookup is performed if the full address lookup did not find a match.	Format: check box Default: unchecked
Blacklist Search Enabled	If enabled, IMSI/MSISDN blacklist lookup is performed prior to the full address lookup.	Format: check box Default: unchecked
FallBack Search Enable	If enabled, Domain-Identifier lookup is performed when the external identifier lookup did not have an exact match.	Format: check box Default: unchecked
Fabr Database Type	This user-configurable parameter takes options as SDS or UDR (Default: SDS). FABR queries are targeted based on field **Fabr Database Type**. Configuration can be changed to choose between SDS or UDR database type. DB change comes into effect after FABR application is disabled/enabled.	Format: field Range: SDS, UDR Default: SDS

3.2.5.2 Editing System Options

Use this task to edit System Options.

1. Click **FABR**, and then **Configuration**, and then **System Options**.
2. Update the relevant fields.

   For more information about each field, see Table 3-5.
3. Click **OK** or **Cancel**.

3.3 Post-Configuration Activities

After FABR configuration is complete, the following activities need to be performed to make FABR fully operational in the system:

- Enabling the FABR application, if it has not already been enabled.
- Status Verification

3.3.1 Enabling the FABR Application

Use this task to enable the FABR application.

1. From each active **SOAM** in a DSR topology, click **Diameter**, and then **Maintenance**, and then **Applications**.
2. Under **DSR Application Name**, select each **FABR** row.

   To select more than one row, press and hold **Ctrl** while you click each row.
3. Click **Enable**.
4. Verify the application status on the page.

   The **Admin State**, **Operational Status**, **Operational Reason**, and **Congestion Level** in each of the selected rows should have changed respectively to **Enabled**, **Available**, **Normal**, and **Normal**.

3.3.2 Status Verification

Use this task to verify FABR status after configuration is complete.

1. Verify Communication Agent (ComAgent) Connection status.
   a. From the active **SOAM** in a DSR topology, click **Communication Agent**, and then **Maintenance**, and then **Connection Status**.
   b. Verify that the **Automatic Connections Count** field displays **X of X in service** where X is the number of peer server connections.
2. Verify server status.
   a. From the active SOAM in a DSR topology, click **Status & Manage**, and then **Server**.
   b. Verify that for each server, the **Appl State** field is **Enabled**, and the **DB**, **Reporting Status**, and **Proc** fields are **Norm**.

3.3.3 Bulk Import and Export

The *Diameter Common User's Guide* describes the use and operation of Bulk Import and Export functions:

- **Help**, and then **Diameter Common**, and then **Bulk Import**
- **Help**, and then **Diameter Common**, and then **Bulk Export**

The Bulk Import and Export functions can be used to export Diameter, IPFE, and Application configuration data in CSV files to a location outside the system, and to import the files (usually edited) into the system where the Import function is executed.

Bulk Import

The Bulk Import operations use configuration data in ASCII Comma-Separated Values (CSV) files (.csv), to insert new data into, update existing data in, or delete existing data from the configuration data in the system.

> **Note:**
>
> Some configuration data can be imported only with the Update operation, and other data can be imported with Insert and Delete operations but not Update. Refer to the *Diameter Common User's Guide* or the **Diameter Common**, and then **Import** Help for valid Import operations.

Import CSV files can be created by using a Bulk Export operation, or can be manually created using a text editor.

> **Note:**
>
> The format of each Import CSV file record must be compatible with the configuration data in the release used to import the file. Across different release versions, column counts may not be compatible, and the import fails.

Files that are created using the Bulk Export operation can be exported either to the local Status & Manage File Management Directory (**Status & Manage**, and then **Files** page), or to the local Export Server Directory.

CSV files that are in the local File Management area can be used for Bulk Import operations on the local system.

Files can be created manually using a text editor; the files must be uploaded to the File Management area of the local system before they can be used for Import operations on the local system.

Multiple Import operations can be performed:

- Insert new configuration data records that do not currently exist in the system
- Update existing configuration data in the system
- Delete existing configuration data from the system

Each Import operation creates a log file. If errors occur, a Failures CSV file is created that appears in the File Management area. Failures files can be downloaded, edited to correct the errors, and imported to successfully process the records that failed. Failures files that are unchanged for more than 14 days and log files that are older than 14 days are automatically deleted from the File Management area.

Bulk Export

The Bulk Export operation creates ASCII Comma-Separated Values (CSV) files (.csv) containing Diameter, IPFE, and Application configuration data. Exported configuration data can be edited and used with the Bulk Import operations to change the configuration data in the local system without the use of GUI pages. The exported files can be transferred to and used to configure another system.

Each exported CSV file contains one or more records for the configuration data that was selected for the Export operation. The selected configuration data can be exported once immediately, or exports can be scheduled to periodically occur automatically at configured times.

Configuration data can be exported in one Export operation:

- All exportable configuration data in the system
- All exportable configuration data from the selected Application, IPFE, or Diameter (each component's data is in a separate file)
- Exportable configuration data from a selected configuration component for the selected Application, IPFE, or Diameter

Exported files can be written to the File Management Directory in the local File Management area (**Status & Manage**, and then **Files** page), or to the Export Server Directory for transfer to a configured remote Export server.

CSV files that are in the local File Management area can be used for Bulk Import operations on the local system.

If the export has any failures or is unsuccessful, the results of the export operation are logged to a log file with the same name as the exported file but with a .log extension. Successful export operations are not logged.

4
Maintenance of FABR

The **Diameter**, and then **Maintenance** GUI provides the FABR specific maintenance functions. In this section describes Admin State, Operational Status, Operational Reason, and Congestion Levels on the **Diameter**, and then **Maintenance**, and then **Applications** page.

4.1 Overview

The FABR application has no maintenance GUI pages of its own. The following Diameter > Maintenance pages provide functions and information that can be used with the FABR application:

- The Diameter > Maintenance > Applications page displays FABR status information including Admin State, Operational Status, and Operational Reason. The page also provides functions to enable and disable the application. See FABR Administrative State and Operational Status and refer to the *Diameter User Guide* and Help for explanations of the page and the status information.
- The Diameter > Maintenance > DA-MPs page displays status and connectivity information for the DA-MP that is running the FABR application. Refer to the *Diameter User Guide* and Help for explanations of the page and the status information.

4.2 FABR Administrative State and Operational Status

The FABR Administrative State (or Admin State) indicates the state that the operator desires the FABR application to be in, and can be manually enabled or disabled. The Operational Status indicates the actual status of the FABR application. The FABR Admin State and Operational Status is updated when the application is started or restarted and when FABR congestion is detected.

Next Generation Network Priority Service (**NGN-PS**) allows National Security/Emergency Preparedness (NS/EP) users to make priority calls/sessions using public networks. The NGN-PS requests are never discarded due to congestion. NGN-PS requests are always processed by FABR application unless FABR application or DP is unavailable, in that case configured Exception Action is used for further routing. For a detailed description of NGN-PS, refer to the *Diameter User's Guide* and Help.

FABR Admin State and Operational Status lists the FABR Admin State and Operational Status related to the DP pool operational status and to FABR congestion levels. It specifies the actions that FABR takes in various situations.

Table 4-1 FABR Admin State and Operational Status

FABR Admin State	DP Operational Status/ Congestion Level	FABR Congestion Level	FABR Operational Status	FABR Actions or Impacts on FABR
Disabled	Any	Any	Unavailable	The default shutdown state

Table 4-1 (Cont.) FABR Admin State and Operational Status

FABR Admin State	DP Operational Status/ Congestion Level	FABR Congestion Level	FABR Operational Status	FABR Actions or Impacts on FABR
Enabled	DP Operational Status = Normal or Degraded DP Congestion Level = Normal OR Minor	Normal/CL1/CL2	Available	FABR receives requests from the DRL. FABR sends queries to the DP.
	DP Operational Status = Normal OR Degraded DP Congestion Level = Major OR Critical OR DP Congestion Abatement in progress	Normal/CL1/CL2	Available	FABR receives requests from the DRL. FABR applies DP Congestion routing exception action.
	DP Operational Status = Normal OR Degraded DP Congestion Level = Any	CL3	Degraded	The DRL stops sending non-NGN-PS requests to FABR. DRL only sends NGN-PS requests to FABR.
	DP Operational Status = Down DP Congestion Level = Any	Any	Unavailable	FABR is shutting down. DRL stops sending requests to FABR.