Signaling Delivery Controller

User Guide

5.2

Catalog Number: RG-022-52-22 Ver. 3

Publication Date: March 2023

Legal Information

Copyright

F5 Inc. (F5) believes the information it furnishes to be accurate and reliable. However, F5 assumes no responsibility for the use of this information, nor any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent, copyright, or other intellectual property right of F5 except as specifically described by applicable user licenses. F5 reserves the right to change specifications at any time without notice.

Trademarks

AskF5, F5, F5 [DESIGN], F5 , OpenBloX, OpenBloX (design), Rosetta Diameter Gateway, Signaling Delivery Controller, SDC, Traffix, and Traffix [DESIGN] are trademarks or service marks of F5, Inc., in the U.S. and other countries, and may not be used without F5's express written consent.

All other product and company names herein may be trademarks of their respective owners.

Patents

This product may be protected by one or more patents indicated at: *http://www.f5.com/about/guidelines-policies/patents*

Confidential and Proprietary

The information contained in this document is confidential and proprietary to F5. The information in this document may be changed at any time without notice.

About F5

F5 (NASDAQ: FFIV) makes the connected world run better. F5 helps organizations meet the demands and embrace the opportunities that come with the relentless growth of voice, data, and video traffic, mobile workers, and applications—in the data center, the network, and the cloud. The world's largest businesses, service providers, government entities, and consumer brands rely on F5's intelligent services framework to deliver and protect their applications and services while ensuring people stay connected. For more information, visit *www.F5.com* or contact us at *Tfx_info@f5.com*.

About this Document

Document Name: F5 Signaling Delivery Controller User Guide

Catalog Number: RG-022-52-22 Ver. 3

Publication Date: March 2023

Document Objectives

This document details and describes the configuration and management procedures of the F5 Signaling Delivery Controller (SDC). This document is designed for end users.

Note: Some of the features described in this document are only supported by virtual or bare-metal installations of the SDC site. Feature support that is dependent on the installation type is mentioned per feature within this document.

Document History

Revision Number	Change Description	Change Location
Ver 2 – June 2021	Update Editing an SNMPv3_Internal User Profile	*Editing an SNMPv3_Internal User Profile*
Ver. 3 – February 2021	1. Updated expected Resolve Session Management behavior 2. Update notification pool guidelines	*1. Table 28: Session Actions* *2. Configuring a Notification Pool*

Conventions

The style conventions used in this document are detailed in Table 1.

Table 1: Conventions

Convention	Use
Normal Text Bold	Names of menus, commands, buttons, user-initiated CLI commands and other elements of the user interface
Normal Text Italic	Links to figures, tables, and sections in the document, as well as references to other documents
`Script`	Language scripts
`Courier`	File names
📄 Note:	Notes which offer an additional explanation or a hint on how to overcome a common problem
Warning: 🔔	Warnings which indicate potentially damaging user operations and explain how to avoid them
🌐	Sections in this guide that relate only to EMS are marked with this icon

Table of Contents

List of Figures

List of Tables

1. Working with the SDC

The F5® Traffix® Signaling Delivery Controller™ (SDC) solution enables routing and exchange of data between different protocols, such as Diameter, SS7, HTTP, and others using an advanced transformation and flow management engine.

The SDC solution can be configured and monitored via a Web UI. In addition, certain configuration and monitoring functionalities are supported by Web Service APIs and a CLI application. For a description of the available Web Service APIs, see the *F5 SDC Web Service API Guide* and for more information about the CLI application, see the *F5 SDC CLI Application Guide*.

2. Getting to Know the SDC/EMS Web UI

The procedures described in this document assume that SDC is remotely configured from a Web Browser. Therefore, in order to perform these procedures, you must have network access to SDC.

The Web UI reflects the two types of sites that can be installed as part of the F5 Traffix SDC solution:

- The SDC Web UI is used to configure peer connectivity and traffic processing definitions for a local SDC site, as well as to configure and view the collected data for the local site.
- The EMS Web UI is used when your deployment includes multiple SDC that you want to manage from a single, centralized site. Using the EMS Web UI, you can perform global configurations, as well as view and monitor your sites’ performance at any given moment, including viewing analytical reports and tracking fault management for troubleshooting and prevention of downtime.

Note: The option to manage multiple SDC sites with an EMS site is only available for bare metal installations.

The SDC and EMS Web UI interface are similar both in their look and feel and in the available actions that can be performed through them. Certain actions, however, are only available in a specific Web UI. Actions that are only supported in the EMS Web UI are marked with this icon:🌐. Such actions include global configurations and viewing certain reports, that are not available using a local SDC Web UI.

1.

2.1 Accessing the SDC/EMS Web UI

This section describes how to access an SDC or an EMS Web UI.

To access an SDC/EMS Web UI:

Launch a web browser.

Note: The SDC/EMS Web UI is supported by the following browsers:

Internet Explorer: Version 1909 OS build 18363.1316

Microsoft Edge: Version 88.0.705.56 (Official build) (64-bit)

Mozilla Firefox: Version 85.0

Google Chrome: Version 88.0.4324.104 (Official Build) (64-bit)

Enter the following HTTP path:

2. **http://<IP address>:8080/MgmtConsole/MgmtConsole.html** in the browser's address line (the IP address that is defined for the Web UI in the site topology file during the installation process). The login screen appears.

Note: The recommended screen resolution is 1280x1024 dpi.

2.2 Logging in to the SDC/EMS Web UI

To successfully log in to the SDC Web UI, the user must authenticate his credentials by performing the following procedure:

1. **To log in to SDC/EMS Web UI:**

2. Enter the **Username** and **Password** provided to you by F5 Systems.

Click **Login**.

Warning: By default, user credentials are authenticated internally by the SDC. This authentication can also be performed using an external LDAP server. To configure the SDC to use external LDAP server, see *Appendix E: Configuring LDAP Authentication*.

If the user authentication process used an external LDAP server, all configuration changes will be logged in the audit log with the LDAP username.

Note: You can view the End User License Agreement, by clicking on **End User License Agreement** in the login screen.

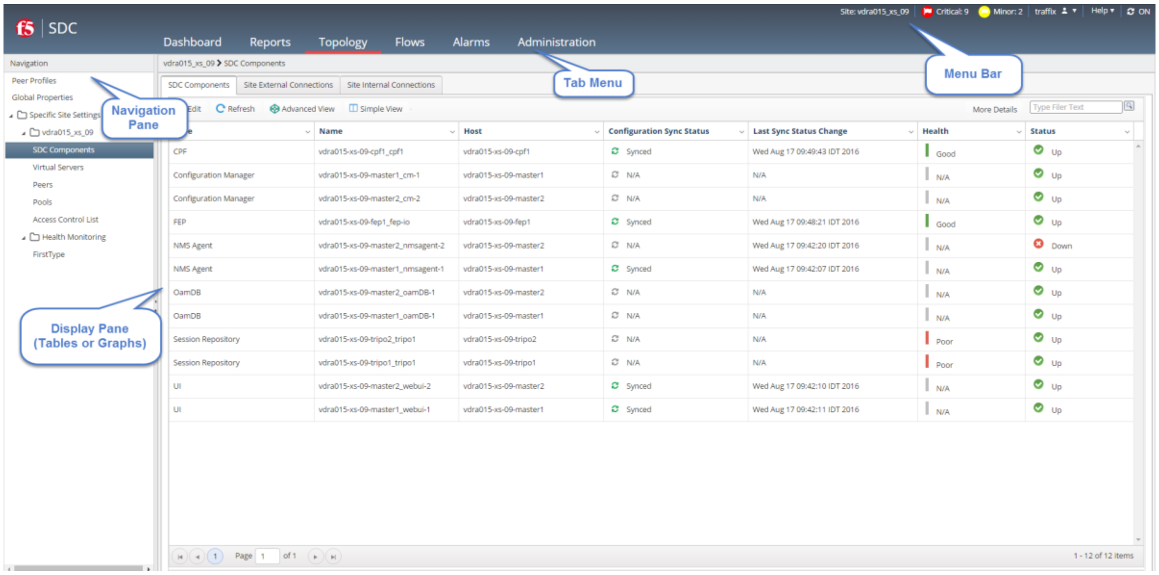
Note: Once you are logged in, if you want to switch to another user, log out and then log in again with the desired user name and password.

2.3 Using the SDC/EMS Web UI

The interface is comprised of the following areas:

- Note: The SDC/EMS Web UI supports Latin-based alphabet characters.
- The Menu Bar
- *The Tab Bar*
- *The Navigation Pane*
- *The Display Pane (tables or graphs)*

Figure 1: SDC Web UI's Interface



e	Name	Host	Configuration Sync Status	Last Sync Status Change	Health	Status
CPF	vdra015-xs-09-cpf1_cpf1	vdra015-xs-09-cpf1	Synced	Wed Aug 17 09:49:43 IDT 2016	Good	Up
Configuration Manager	vdra015-xs-09-master1_cm-1	vdra015-xs-09-master1	N/A	N/A	N/A	Up
Configuration Manager	vdra015-xs-09-master2_cm-2	vdra015-xs-09-master2	N/A	N/A	N/A	Up
FEP	vdra015-xs-09-fep1_fep-io	vdra015-xs-09-fep1	Synced	Wed Aug 17 09:48:21 IDT 2016	Good	Up
NMS Agent	vdra015-xs-09-master2_nmsagent-2	vdra015-xs-09-master2	N/A	Wed Aug 17 09:42:20 IDT 2016	N/A	Down
NMS Agent	vdra015-xs-09-master1_nmsagent-1	vdra015-xs-09-master1	Synced	Wed Aug 17 09:42:07 IDT 2016	N/A	Up
OamDB	vdra015-xs-09-master2_oamDB-1	vdra015-xs-09-master2	N/A	N/A	N/A	Up
OamDB	vdra015-xs-09-master1_oamDB-1	vdra015-xs-09-master1	N/A	N/A	N/A	Up
Session Repository	vdra015-xs-09-tripo2_tripo1	vdra015-xs-09-tripo2	N/A	N/A	Poor	Up
Session Repository	vdra015-xs-09-tripo1_tripo1	vdra015-xs-09-tripo1	N/A	N/A	Poor	Up
UI	vdra015-xs-09-master2_webui-2	vdra015-xs-09-master2	Synced	Wed Aug 17 09:42:10 IDT 2016	N/A	Up
UI	vdra015-xs-09-master1_webui-1	vdra015-xs-09-master1	Synced	Wed Aug 17 09:42:11 IDT 2016	N/A	Up

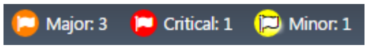
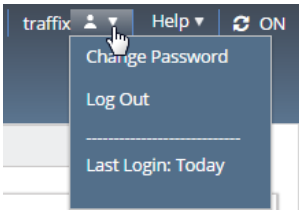
1 - 12 of 12 items

Note: The SDC/EMS Web UI supports Latin-based alphabet characters.

2.3.1 The Menu Bar

Table 2 describes the SDC/EMS menu tabs.

Table 2: The Menu Bar

Tab	**SDC Description**	🌐 **EMS Description**
	Enables you to view generated traps in the Trap Viewer table per trap severity level	Enables you to view generated traps in the Trap Viewer table per trap severity level
🌐	N/A	Shows that you are working with an EMS to manage multiple sites
 traffix Help ▾ ON Change Password Log Out Last Login: Today	Enables you to change your user interface information	Enables you to change your user interface
Change Password	Enables you to change your password to access the user interface.	Enables you to change your password to access the user interface.
Log Out	Enables you to log out from the user interface, and returns you to the login screen.	Enables you to log out from the user interface, and returns you to the login screen.
Help	Enables you to access the SDC Web UI HTML Help or API scripting	Enables you to access the EMS Web UI HTML Help or API scripting
	Auto Refresh is enabled by default so that the most updated configuration data is displayed. Automatic refresh can be enabled or disabled	**Auto Refresh** is enabled by default so that the most updated configuration data is displayed. Automatic refresh can be enabled or disabled

2.3.2 The Tab Bar

Table 3 describes the SDC/EMS tab bar.

Table 3: The Tab Bar

Tab	SDC Description	🌐 EMS Description
Dashboard	Displays current system KPI’s, statistics graphs and recently generated SNMP traps.	Displays current system KPI’s, statistics graphs and recently generated SNMP traps.
Reports	System wide reports and graphs with optional filtering for both statistics and short-term tracing.	System wide reports and graphs with optional filtering for both statistics and short-term tracing.
Topology	Provides topology entity configuration interface.	Displays a bird-eye topology view and provides topology entity configuration interface.
Flows	Provides contextual flow management editing interface	Provides contextual flow management editing interface (when EMS is installed Routing is globally configured).
Alarms	Provides an overview of SNMP active alarms and an alarm history log	Provides an overview of SNMP active alarms and an alarm history log
Administration	Provides an interface for administrative procedures such as user management, backup and restore, global threshold management, list definitions, etc.	Provides an interface for administrative procedures such as user management, backup and restore, global threshold management, list definitions etc.

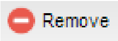
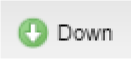
2.3.3 The Navigation Pane

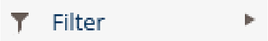
The Navigation Pane displays the sub-menu options for each of the tabs.

2.3.4 Common Actions

This section describes the common actions that are available to a user through the Web UI. Users can easily select an entity in a table and then make changes to it, such as adding, a peer or pool. Some of the actions are available through the column's context menu.

Table 4: Common Actions

Button	Description
Submit	Saves changes applied to a selected item.
Add	Adds an item.
Edit...	Edits the selected item.
Enable	Sets the **Administrative State** to Enabled for the selected item.
More Details	Expands the **Topology** > **SDC Components/Peers/Pools** and Monitoring tables for horizontal scroll view of additional data Less Details removes the display of the additional data. Expands some decision tables (such as, **Flows** > **Routing Rules/Transformation and Administration** > **Tracing/Message Prioritization**) with a Comment column.
Disable	Sets the **Administrative State** to Disabled for the selected item
Duplicate	Creates another item in the table with all the definitions of the selected item.
 Remove	Removes the selected item.
Move To	Moves the selected row to the first or last row of a selected page, as defined in the Move row window.
 Down	Moves the selected item to a lower place in the list.
Up	Moves the selected item to a higher place in the list.

Button	Description
	Defines the attributes (AVP’s) of the rule table.
	Displays all the configured rules for flow management.
	Displays the selected rule in script language.
	Refreshes the selected item’s properties in certain screens, in case they were modified by another user in a remote location.
	Sorts the table in an alphabetically ascending order.
	Sorts the table in an alphabetically descending order.
	Selects which table columns to display.
	Selects which table rows to display according to a filter: rows which match the column’s filter text are displayed.

Note: The buttons availability changes according to the selected item in the Navigation pane (e.g.: when an item cannot be moved, the Down/Up buttons are unavailable).

2.3.5 Keyboard Navigation

In this SDC release, there is an option to navigate through the Web UI using your keyboard. This option is supported in all SDC and EMS screens.

The navigation keys and their corresponding actions are detailed in *Table 5*.

Table 5: Keyboard Navigation

Key	Action
Tab	Moving forward to the next element or section
Shift + Tab	Moving backwards to the previous element or section
Alt + J	Brings focus to Tab Bar
Alt + M	Brings focus to Menu Bar

Key	Action
Alt + N	Brings focus to Navigation Pane
Alt + Arrow Down	Opening a drop-down list
Shift + Arrow Down/Arrow Up	Moves the items in the drop list without having to open it
Arrow Up/Arrow Down	1. Navigation between drop-down list items 2. Moves the screen view focus up and down 3. Navigates through the navigation panel tree
Arrow Left/Arrow Right	Opens and close navigation folders when standing on them
Space	Selects peers/pools. Pressing the space bar again will cancel the selection
Space + Enter + Tab	Selects and saves the peer/pool selection
Enter	1. Selecting an element from a drop-down list 2. Selecting the tab in focus 3. Accessing a link 4. Opening the information detail when standing on the ? icon
Alt + A	Add flow
Alt + R	Delete flow
Ctrl + Q	Go out from script editor
Alt + W	Returns the focus to tab
Alt + left arrow	Go back to previous screen
Home/End	Brings focus to the beginning or end of the Navigation Pane

2.3.6 SDC Decision Tables

The SDC decision tables are used to define message-specific behavior for the following decisions:

- Dynamic Peer Profiles

- Flows
- Transformation (pre and post)
- Session Management
- Routing Rules
- Message Prioritization
- Tracing

Decision tables are tables of rules, defined by the user. These rules are used to apply specific actions and configurations to certain messages only, based on the message properties.

Note: The SDC supports up to 1000 routing rows in a decision table.

Each rule is configured with conditions that filter messages according to specific message properties, as well as a rule action that contains a specific action and/or configuration that will be applied to messages that meet the rule conditions. Each rule is defined with three parameters – the rule name, the rule attributes and the rule action.

The **rule name** is displayed in the ID column in the decision table. It is configured by the system and is made up of a pre-defined prefix (per decision table type) and the rule number.

The **rule attributes** are each displayed in a dedicated column with their user-defined name in the decision table. They are configured by the user, and when no rule attributes are configured for the decision table, only the rule name and rule action columns appear in the decision table. The rule attributes are message properties that are used as conditions to filter the incoming messages to match a specific rule.

The **rule action** is displayed in a column in the decision table. When the rule action is configured by script, it is not displayed in the decision table, but rather in the area below the table when a row in the table is selected.

Note: There are some rule actions that have associated rule configurations. The associated rule configurations are displayed in the area below the table when the row in the table is selected.

When a message is received by the SDC, its properties are compared against the values defined for the rule attributes for the rule that appears in the first row in the decision table. If all the defined rule attribute values are matched, the actions defined for that rule and its associated configurations (when applicable) are implemented for the message. If the rule's criteria are not all matched, the rule attribute values in the next row in the decision table are checked, until a rule is found with all the matching criteria.

Configuring the decision tables includes the following procedures:

- *Adding Rule Attributes*
- *Defining the Rule Criteria*
- *Creating and Defining Attribute Lists*

2.3.6.1 Adding Rule Attributes

Rule attributes are message properties that are used as the rule's criteria. Each rule attribute must be added to the decision table by the user. Once a rule attribute is added to the decision table, you can define the rule attribute value for each rule in the decision table.

To add a rule attribute to a decision table:

1. In the decision table screen, click **Rule Attributes**.
2. In the Rule Attributes window, click **Add**. A new row is added to the Rule Attribute table.
3. In the **Label** column, enter a name for the rule attribute. This name will appear as the column name for this rule attribute in the decision table.

Note: Label cell entries cannot start with a number.

In the **Attribute** column, enter the message property that is checked against the defined value for this attribute in the decision table.

4. Note: The SDC has a list of predefined properties for various SDC entities that can be used in any of the decision tables. For information about the predefined properties, see *Appendix D: Decision Table Attributes*.

5. In the **Type** column, from the drop-down list, select the way that the message property is checked against the value defined for this attribute.

Table 6: Type Description

Filter Type	**Checks That…**
STRING	The string that the user defines for this rule attribute in the decision table matches the string data retrieved from the message.
STRING COMPLEMENT	The string that the user defines for this rule attribute in the decision table does not match the string data retrieved from the message.
SUBSTRING	The substring that the user defines for this rule attribute in the decision table matches a substring of the string data retrieved from the message. Note: Using this filter type may impact performance. It is therefore recommended to use the prefix or suffix filter types.
SUBSTRING COMPLEMENT	The substring that the user defines for this rule attribute in the decision table does not match a substring of the string data retrieved from the message. Note: Using this filter type may impact performance. It is therefore recommended to use the prefix or suffix filter types.
SUBSTRING IGNORE CASE	The substring that the user defines for this rule attribute in the decision table matches a substring of the string data retrieved from the message, without taking upper/lower case differences into account. Note: Using this filter type may impact performance. It is therefore recommended to use the prefix or suffix filter types.

Filter Type	Checks That…
SUBSTRING IGNORE CASE COMPLEMENT	The substring that the user defines for this rule attribute in the decision table does not match a substring of the string data retrieved from the message, without taking upper/lower case differences into account. Note: Using this filter type may impact performance. It is therefore recommended to use the prefix or suffix filter types.
OCTET STRING	The string that the user defines for this rule attribute in the decision table matches the binary data retrieved from the message. Note: An IP address must be in hexadecimal syntax without the “Ox” prefix.
SCRIPT	The script that the user defines for this rule attribute returns “true”.
REGEXP	The regular expression that the user defines for this rule attribute in the decision table matches the string data retrieved from the message. Note: Using this filter type may impact performance and is therefore not recommended.
NUMBER	The number that the user defines for this rule attribute in the decision table matches the numerical data retrieved from the message.
NUMBER COMPLEMENT	The number that the user defines for this rule attribute in the decision table doesn’t match the numerical data retrieved from the message.
POSITIVE NUMBER	The range that the user defines (between 1 and a maximum value) for this rule attribute in the decision table matches the numerical data retrieved from the message.
BOOLEAN	The Boolean value that the user defines for the rule attribute in the decision table matches the Boolean value retrieved from the message.
NUMBER RANGE	The number range that the user defines for this rule attribute in the decision table encompasses the numerical data retrieved from the

Filter Type	Checks That…
	message. Guidelines: Use the following format to define the range: "<min> - <max>", where <min> is the value equal to the lowest range value and <max> is the value equal to the highest range value.
STRING RANGE	The string representation of a number range that the user defines for this rule attribute in the decision table encompasses the numerical data retrieved from the message. Guidelines: Use the following format to define the range: "<min> - <max>", where <min> is the value equal to the lowest range value and <max> is the value equal to the highest range value.
DATE RANGE	The date range that the user defines for this rule attribute in the decision table encompasses the date data retrieved from the message. Guidelines: Use the following format to define the range: "<min> - <max>", where <min> is the value equal to the lowest range value and <max> is the value equal to the highest range value. Note: The date format is dd.mm.yyyy
PEER STATE	The peer state that the user defines for this rule attribute in the decision table matches the peer data retrieved from the message.
PROTOCOL	The protocol that the user defines for this rule attribute in the decision table matches the protocol data retrieved from the message.
IP ADDRESS	The IP address that the user defines for this rule attribute in the decision table matches the IP address data retrieved from the message. Note: The value can either be a machine name or a textual representation of its IP address.
IP ADDRESS RANGE	The IP address range that the user defines for this rule attribute in the decision table encompasses the IP address data retrieved from the message. Guidelines: Use the following format to define the range: "<min> - <max>", where <min> is the value equal to the lowest range value and <max> is the value equal to the highest range value.

Filter Type	Checks That…
	Note: The minimum and maximum values can either be a machine name or a textual representation of its IP address.
STRING IGNORE CASE	The string that the user defines for this rule attribute in the decision table matches the string data retrieved from the message, without taking upper/lower case differences into account.
STRINGE IGNORE CASE COMPLEMENT	The string that the user defines for this rule attribute in the decision table does not match the string data retrieved from the message, without taking upper/lower case differences into account.
POOL STATE	The pool state that the user defines for this rule attribute in the decision table matches the pool data retrieved from the message.
TIME	The time that the user defines for this rule attribute in the decision table matches the time data retrieved from the message. Note: The time format is hh.mm.ss
TIME RANGE	The time range that the user defines for this rule attribute in the decision table encompasses the time data retrieved from the message. Guidelines: Use the following format to define the range: “<min> - <max>”, where <min> is the value equal to the lowest range value and <max> is the value equal to the highest range value. Note: The time format is hh.mm.ss
DATE	The date that the user defines for this rule attribute in the decision table matches the date data retrieved from the message. Note: The date format is dd.mm.yyyy
PREFIX	The string that the user defines for this rule attribute in the decision table matches the prefix of the string data retrieved from the message.
PREFIX IGNORE CASE	The string that the user defines for this rule attribute in the decision table matches the prefix of the string data retrieved from the message, without taking upper/lower case differences into account.

Filter Type	Checks That…
PREFIX COMPLEMENT	The string that the user defines for this rule attribute in the decision table does not match the prefix of the string data retrieved from the message.
PREFIX IGNORE CASE COMPLEMENT	The string that the user defines for this rule attribute in the decision table does not match the prefix of the string data retrieved from the message, without taking upper/lower case differences into account.
SUFFIX	The string that the user defines for this rule attribute in the decision table matches the suffix of the string data retrieved from the message.
SUFFIX IGNORE CASE	The string that the user defines for this rule attribute in the decision table matches the suffix of the string data retrieved from the message, without taking upper/lower case differences into account.
SUFFIX COMPLEMENT	The string that the user defines for this rule attribute in the decision table does not match the suffix of the string data retrieved from the message.
SUFFIX IGNORE CASE COMPLEMENT	The string that the user defines for this rule attribute in the decision table does not match the suffix of the string data retrieved from the message, without taking upper/lower case differences into account.
LIST	One of the values included in the list that the user defines for this rule attribute in the decision table matches the data retrieved from the message. Note: The list name cannot be edited. In addition, lists that are associated with a configured rule cannot be deleted. Note: The list filter type can be used up to 3 times in a decision table.
LIST COMPLEMENT	None of the values included in the list that the user defines for this rule attribute in the decision table match the data retrieved from the message. Note: The list name cannot be edited. In addition, lists that are associated with a configured rule cannot be deleted.

Filter Type	Checks That…
	Note: The list filter type can be used up to 3 times in a decision table.
LIST PREFIX	The string (message AVP prefix) that the user defines for this rule attribute in the decision table matches one of the strings defined in the list.
LIST PREFIX IGNORE CASE	The string (message AVP prefix) that the user defines for this rule attribute in the decision table matches one of the strings defined in the list, without taking upper/lower case differences into account.
LIST PREFIX COMPLEMENT	The string (message AVP prefix) that the user defines for this rule attribute in the decision table does not match one of the strings defined in the list.
LIST PREFIX IGNORE CASE COMPLEMENT	The string (message AVP prefix) that the user defines for this rule attribute in the decision table does not match the prefix of the list data retrieved from the message, without taking upper/lower case differences into account.
LIST SUFFIX	The string (message AVP suffix) that the user defines for this rule attribute in the decision table matches one of the strings defined in the list.
LIST SUFFIX IGNORE CASE	The string (message AVP suffix) that the user defines for this rule attribute in the decision table matches one of the strings defined in the list, without taking upper/lower case differences into account.
LIST SUFFIX COMPLEMENT	The string (message AVP suffix) that the user defines for this rule attribute in the decision table does not one of the strings defined in the list.
LIST SUFFIX IGNORE CASE COMPLEMENT	The string (message AVP suffix) that the user defines for this rule attribute in the decision table does not match one of the strings defined in the list, without taking upper/lower case differences into account.

6. In the **Description** column, enter a free text description of the attribute.

7. Repeat steps 2-6 until all rule attributes have been added.

Click **Save**. The decision table is now updated with columns reflecting the label values of the added rule attributes.

2.3.6.2 Defining the Rule Criteria

8.

The values defined for each Rule Attribute ensure that each message is correctly processed by the SDC. Only once all the defined rule attribute values are matched is the rule action implemented for the message.

To define rule attribute values:

1. In the decision table screen, click **Add**. A new row is added to the decision table with the corresponding prefix the next available serial number.

2. Fill in the value field for each rule attribute as follows:

a. A value based on the rule attribute type (string, boolean, etc.) – the message and entity will be checked to see if they contain the property with the matching value according to the filter type (as defined when *Adding Rule Attributes*).

Note: If the rule attribute is defined as a list, enter the name of the list as it appears in the Attribute Lists table. For more information, see *Creating and Defining Attribute*

b.*Lists*.

c. A value of “**~exists**” – when this value is entered, the message and entity are checked to see if they contain the property. The rule attribute will be approved as long as the property exists, irrelevant of the property value.

A value of “**~none**” – when this value is entered, the message and entity will be checked to see if they contain the property. The rule attribute will only be approved if the property **does not** appear in the message and entity.

2.3.6.2.1 Creating and Defining Attribute Lists

Before a list can be used as a rule attribute, it must be created and populated with values.

To create a list:

1. Go the **Administration** > **Attribute Lists**.

   The table displays all configured lists by name and shows a user-defined description (if one was entered).

2. Click **Add**. The **Add Attribute List** window appears.

3. In the **List Name** field, enter a name for the list.

   Note: The name can contain a maximum of 50 characters.

4. In the **List Description** field, enter a short description of the list. This value will be displayed in the Attribute Lists table.

   Note: The description can contain a maximum of 250 characters.

5. Populate the list with the desired values. These values can be entered manually, or imported from a CSV file.

   Note: Each list can contain a maximum of 1000 values.

   Note: Each value can contain a maximum of 250 characters.

   a. To manually add the list values:

      i. Verify that the **Add Items Manually** option is selected for the **Items** field.

      ii. Click **OK**. The list is added to the Attribute Lists table.

      iii. Select the list in the Attribute Lists table. A table with the list name appears at the bottom pane, with any previously defined values.

      iv. In the **List Name** table, click **Add**.

      v. In the new row that has appeared in the table, enter the desired value.

vi. Click **Add** to enter additional values.

The list is now populated with the added values.

To import the values from an CSV file:

i. Select **Import Items From File**.

b. ii. In the **File** field, enter the path to the file, or click **Browse** and select the file.

iii. Click **OK**. The list is added to the Attribute Lists table, with the defined value from the imported file.

Click **Submit**.

6.

2.3.6.2.2 Importing and Exporting Attribute Lists

The values defined for an attribute list can be imported from and exported to a CSV file.

To export the values defined for an attribute list:

1. In **Administration** > **Attribute Lists**, select the list. The defined attribute list values are displayed in the **List Name** table in the bottom pane.

2. In the **List Item** table, click **Export**. A CSV file with the defined values, with the list name (<List_Name>.csv) is created.

1. **To import the values defined for an attribute list:**

2. In **Administration** > **Attribute Lists**, select the list. The defined attribute list

3. values are displayed in the **List Item** table in the bottom pane.

4. In the **List Item** table, click **Import**. An Import Row(s) window appears.

In the **File** field, enter the path to the file, or click **Browse** and select the file.

Click **OK**. The list is now defined with the values from the file.

Note: This action replaces any existing values defined for the list.

2.3.6.3 Defining Rule Actions and Configurations

The Rule Actions defined for each rule in the decision table detail how a message matching the rule criteria will be processed. For more information about each decision table, refer to the appropriate section in this guide.

3. Configuring the SDC Topology

This chapter describes how you configure and view the SDC topology, encompassing the different network entities connected to the SDC site.

3.1 Peer Profiles

Peer Profiles are rules according to which you may choose to handle specific Remote Peers. When a Remote Peer is assigned a Peer Profile, you may choose to send it unique messages or accept/reject it (using the Access Control List). For information about configuring the Peer Profiles, see *Configuring Peer Profiles*.

Note: When EMS is installed, Peer Profiles are globally configured. When only SDC is installed, they are locally configured.

3.2 Global Properties

The Global Properties menu option provides you the opportunity to define property values to use in scripts relating to all SDC related objects. Once defined, using these properties in scripts will reflect the specified value.

1. **To add a global property:**

2. Go to **Topology** > **Global Properties** > **Add**.

3. In the **Name** field, enter a user friendly property name.

4. In the **Value** field, enter the desired value for the property.

In the **Path** field, the file path to the property definition is displayed.

5.

Note: The path name is only displayed once the changes are submitted.

Click **Submit**.

Note: Global properties can also be defined using the setEntityProperties Web Service API method and retrieved using the getEntityProperties Web Service API method or using

Groovy scripting. For more information about the Web Service API methods, see *F5 SDC Web Services API Guide*.

3.2.1 Specific Site Settings

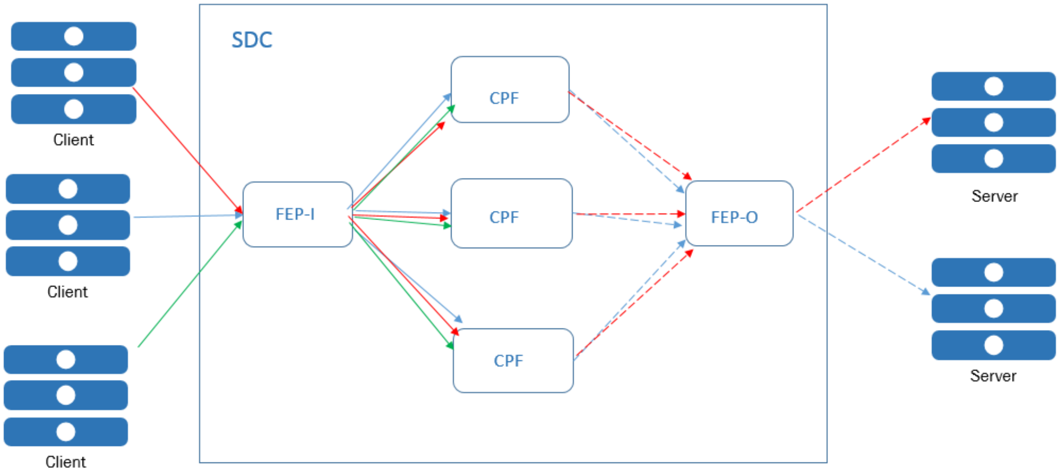
This section describes the different components that are configured per site.

3.2.1.1 SDC Components

SDC comprises the hardware and software required to handle high traffic load and provide high availability. A single instance of SDC application, run on a designated hardware and is comprised of two types of components - FEP (Front-End Proxy) and CPF (Control Plane Function) - which share the same framework. FEP constructs a transport pipeline with each of its Diameter peers. All FEP nodes are connected to all CPF nodes. When a new CPF node joins the cluster, all FEP nodes connect to it. When a new FEP node joins the cluster it automatically connects to all CPF nodes.

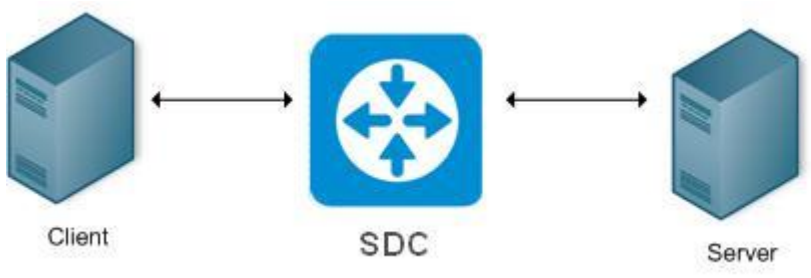
Figure 2 shows the basic network architecture:

Figure 2: Network Architecture



The combination of the two components, CPF and FEP comprises SDC:

Figure 3: SDC, Client and Server



SDC Components are defined throughout the SDC installation procedure. Each site that SDC is installed in must have at least one SDC Component. Each SDC Component is associated with a single or multiple IP Address, a port number through which it operates and the network protocols it supports. The IP address that represents the SDC Component is usually mapped to multiple servers. In these cases, SDC must verify the availability of all servers associated with the SDC Component and distribute traffic across all actual servers. When doing so, it also translates the SDC's IP address to the actual server's IP address and the SDC Component's port number to the actual server's port number. For information about configuring the SDC components, see *Configuring a Site's User Properties*.

3.2.1.2 Virtual Servers

Virtual Servers are virtual instances of SDC used to facilitate every protocol used by SDC to communicate with dynamic peers (Clients). Traditionally, a single Virtual Server represents each protocol that the SDC Component listens to in the network. For information about configuring the virtual servers, see *Configuring Virtual Servers*.

3.2.1.3 Remote Peers

Remote Peers are clients (AAA service consumers) and servers (AAA service providers) that are linked to SDC Components. Throughout SDC service providing procedure, information is sent to the Remote Peers or received from them.

A Remote Peer is combined of an IP address/s and a port number through which it operates, and the protocol in which it operates. Several Remote Peers may be hosted on a single hosting machine. For information about configuring the remote peers, see *Configuring Peers*.

3.2.1.4 Pools

Pools are groups of peers. Peers are grouped together in a pool in order to make the administrator’s work more efficient. Pools allow the administrator to assign a single common policy to multiple servers. When a request is sent, it is associated with an SDC Component that is linked to a group of remote peers. SDC uses the pool configuration in order to decide how to approach the load balancing and translation procedures.

Each pool is identified by its name and is assigned with a single policy. After creating a pool, naming it, adding peers to it and selecting its policy, it can be modified at any given moment. For example: you may change the pool’s name, add new peers to it, or remove existing ones from it. You may also change the policy assigned to the pool.

Pools are independent. This means that they can be added and configured in the SDC system without being associated with the SDC Component. However, if an SDC Component is not associated with the peers in the pool, SDC will not use the pool during load balancing and translation service performance, upon request retrieval. Each remote peer may be associated to several pools. Pools can also be filled automatically by assigning a peer profile to the Pool. For information about configuring the pools, see *Configuring Pools*.

3.2.1.5 Access Control List

The Access Control List allows you to compose rules that determine which Client Peers are accepted by SDC and which are rejected by it. Client Peers are identified by their IP address or peer profile. An accepted Client Peer may send requests to a Server Peer, while a rejected Client Peer may not. For information about configuring the access control list, see *Configuring the Access Control List*.

3.2.2 The Control Plane Traffic Flow – SDC's Services

The control plane traffic flow is transparent to the end user. The most common traffic flow is the one in which requests are transmitted from the Remote Peer (AAA Client) to SDC and from SDC to a Server Peer (AAA Server). But since each SDC is usually associated with more than one actual server, this is not the only optional flow.

When a Remote Peer sends a request, it is sent to the SDC's Address. If the SDC's address is mapped to several actual servers, SDC maps the request to an available Server Peer associated to it, according to the SDC algorithm. When an answer is sent back to the Remote Peer, the source and destination addresses are reversed so that the answer reaches the right destination.

3.3 Configuring the Topology

This section introduces how to create and configure the different topology nodes of the SDC – SDC Components, Virtual Servers, Peers and Pools.

3.3.1 Configuring Peer Profiles

Peer Profiles are logical objects used to tag Peers. Peer Profiles may be assigned Association Rules with which the Peers are compared. When an unknown Peer matches the association rule, it is tagged. Tagged Peers may send or receive unique messages. Peer Profiles may also be used as an additional filtering parameter in *Configuring the Access Control List.*

You can do the following actions as part of configuring peer profiles:

- *Viewing the List of Peer Profiles*
- *Adding a Peer Profile*

Note: When EMS is installed, Peer Profiles are globally configured. When only SDC is installed, they are locally configured.

3.3.1.1 Viewing the List of Peer Profiles

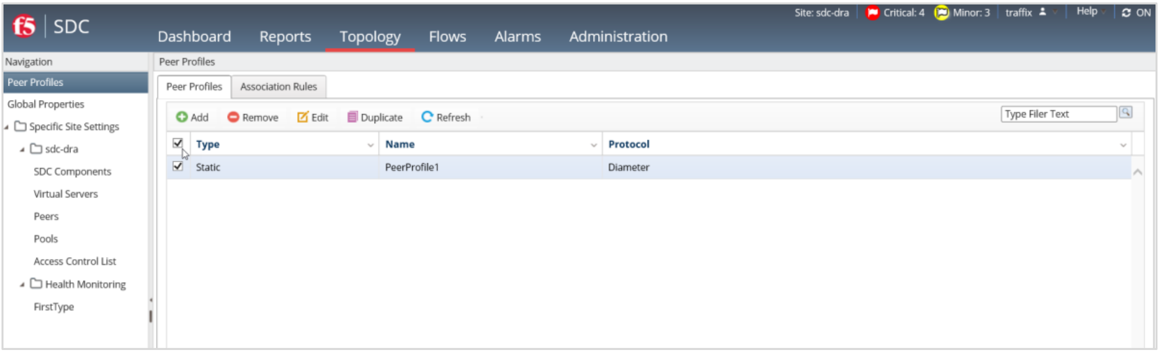
You can view the list of available peer profiles.

To view the list of Peer Profiles:

Go to **Topology** > **Peer Profiles**.

Figure 4: Peer Profiles

1.



☑	Type	Name	Protocol
☑	Static	PeerProfile1	Diameter

Table 8 presents a list of peer properties:

Table 7: Peer Profile’s Properties

Column	Description
Type	Shows peer profile as static (client or server) or dynamic (unselected)
Name	A user-friendly display name assigned to the Peer Profile. e.g. PeerProfile1
Protocol	The signaling protocol used by the Peer. Profile e.g. Diameter

3.3.1.2 Adding a Peer Profile

In addition to adding a peer profile, you can also edit existing peer profiles by selecting a peer profile, clicking **Edit**, and then select the relevant tab and parameters as described in this section. The specific tabs and parameters vary slightly depending on which peer profile protocol you select. The specific wizard configurations per protocol follow a description of the **General** and **User Properties** wizard configurations that are for each peer profile protocol.

To add a new Peer Profile:

Go to **Topology** > **Peer Profiles** Click **Add**. The Add Peer Profile wizard is displayed:

1. For Type, select **Static** to create a static (client or server) Peer Profile or **Dynamic** to create a (client) dynamic Peer Profile or **Peer Profiles**.

2. In the **Name** field, enter a user-friendly display name to identify the Peer Profile. e.g. PeerProfile1. The name should be a meaningful name, as it is used to help the
3. user to distinguish between different profiles based on one of the properties of all the peers which share this profile, e.g. – GGSN clients, or servers from specific data center.

Note: The name should not include special characters, such as # $ %, that are not XML compliant. Once saved, you cannot edit the peer profile name.

Geo-redundant operators with two MMEs should configure two different peer profiles for each MME.

---

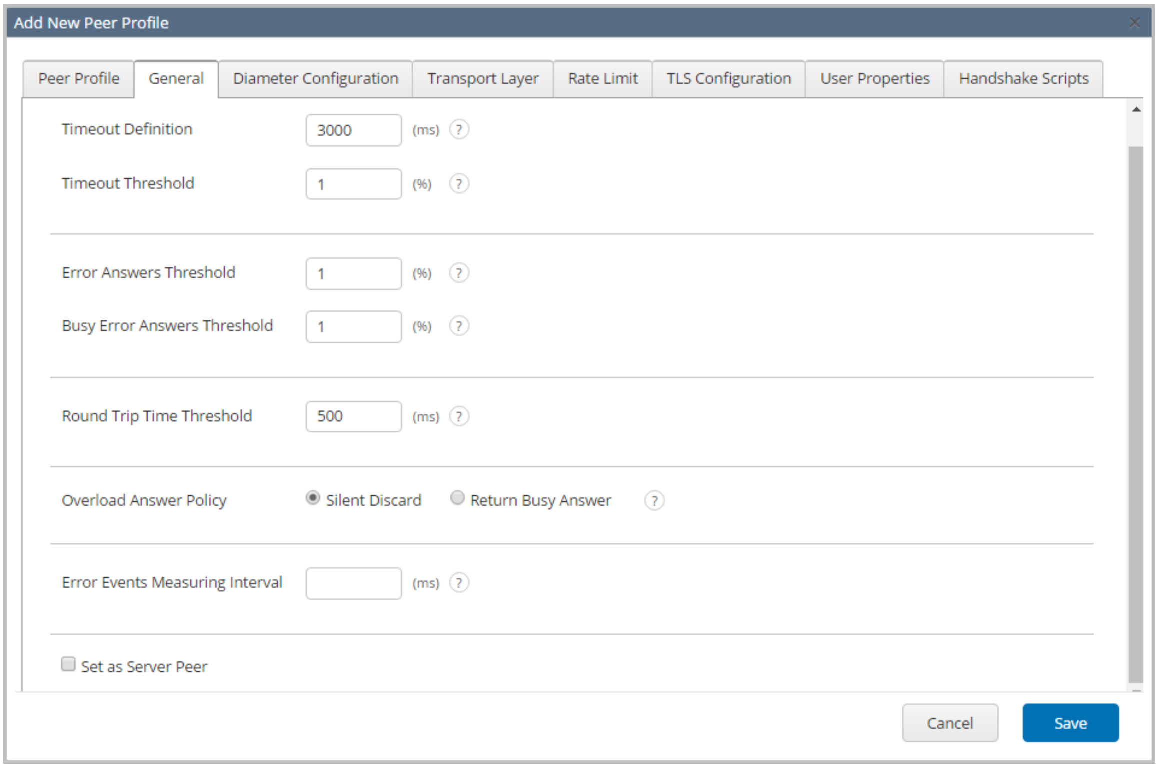
4. In the **Protocol** field, select the signaling protocol used by the Peer Profile, e.g. Diameter.

---

5. Note: The SS7 protocol is only supported in bare-metal deployments. After selecting the SS7 protocol, verify that the selected option for Type is “Dynamic”.

Under the **General** tab (available to all protocols):

Figure 5: Peer Profile Configuration



Timeout Definition: 3000 (ms)

Timeout Threshold: 1 (%)

Error Answers Threshold: 1 (%)

Busy Error Answers Threshold: 1 (%)

Round Trip Time Threshold: 500 (ms)

Overload Answer Policy: ● Silent Discard ○ Return Busy Answer

Error Events Measuring Interval: (ms)

□ Set as Server Peer

a. In **Timeout Definition**, set the time frame (in milliseconds) in which the peer is expected to answer requests.

Note: When configured in routing (**Flows** > **Flows** > **Routing**>**Rule Configuration** > **Max Resend Attempts**), the request is resent if the defined timeout expires before the peer sends an answer. The default is set to three seconds. Timed-out requests are counted for determining a server peer’s health. For additional information on Health Monitoring, see *Health Monitoring*.

b. In **Timeout Threshold,** set the allowed ratio between the number of requests sent to the peer and the number of requests not answered by the peer in the defined timeout period.

Note: This indicator is used for determining a server peer's health and overload control. For more information about monitoring a peer's health, see *Viewing the List of Peers* and for overload control, see *Configuring Overload Control Policy*.

c. **In Error Answers Threshold,** set the allowed ratio between the number of requests sent to the peer and the number of error answers returned from the peer.

Note: This indicator is used for determining a server peer's health. For more information about monitoring a peer's health, see *Viewing the List of Peers*

d. For Diameter and HTTP protocols, in **Busy Error Answers Threshold**, set the allowed ratio between the number of requests sent to the peer and the number of error answers (with a DIAMETER_TOO_BUSY 3004 result code or a 503/505 HTTP result code) returned from the peer.

e. In **Round Trip Time Threshold** set the threshold (in milliseconds) for the allowed round trip time frame.

Note: This indicator is used for determining a server peer's health and overload control. For more information about monitoring a peer's health, see *Viewing the List of Peers* and for overload control, see *Configuring Overload Control Policy*.

f. In Overload Answer Policy,

- **Silent Discard** – discards all incoming messages and does not return any answer.

Note: This is the default behavior.

- **Return Busy Answer** –returns the request to the SDC and a BUSY Remote Node Event is sent

In **Error Events Measuring Interval**, set the time frame in which error detecting procedure is performed.

g. Note: This parameter can be used in the health monitoring script to define the time frame in which an error detecting procedure is performed.

Select **Set as Server Peer** for server only peer profiles.

Under the **User Properties** tab (available to all protocols):

h.

6. You can create additional properties for the Peer Profile and define the value for these properties. These properties can be used in the Peer Profile scripts and decision table.

Click **Add**.

a.

In the **Name** field, enter a user friendly property name.

b.

c. In the **Value** field, enter the desired value for the property.

d. In the **Path** field, the path name for the property is displayed.

Note: The path name is only displayed once the peer is added.

User properties can also be defined using the setEntityProperties Web Service API method and retrieved using the getEntityProperties Web Service API method or using Groovy scripting. For more information about the Web Service API methods, see the *F5 SDC Web Services API Guide*.

3.3.1.3 Assigning an Association Rule to a Dynamic Peer Profile

1. Peer Profiles may be assigned Association Rules with which the Peers are compared.

2.

To assign an association rule:

Go to **Topology** > **Peer Profiles** > **Association Rules** > **Add**.

Under the **Dynamic Peer Profile** column, select one of the relevant configured dynamic peer profiles.

Configure rule attributes for the peer profile rules by following the instructions in *Adding Rule Attributes.*

Click **Submit**.

3.

3.3.1.4 Configuring Multiple Connections per Peer

4. The SDC supports multiple connections between one dynamic client peer and the FEP. The client peer needs to be configured with a peer profile that invokes a Capabilities Exchange Answer script. Once this script is invoked, each new connection of the same peer is considered as a new peer with the same Origin Host name followed by a suffix (SN_<index #>).

Note: Up to 50 connections are supported and any connections not being used, can be reapplied.

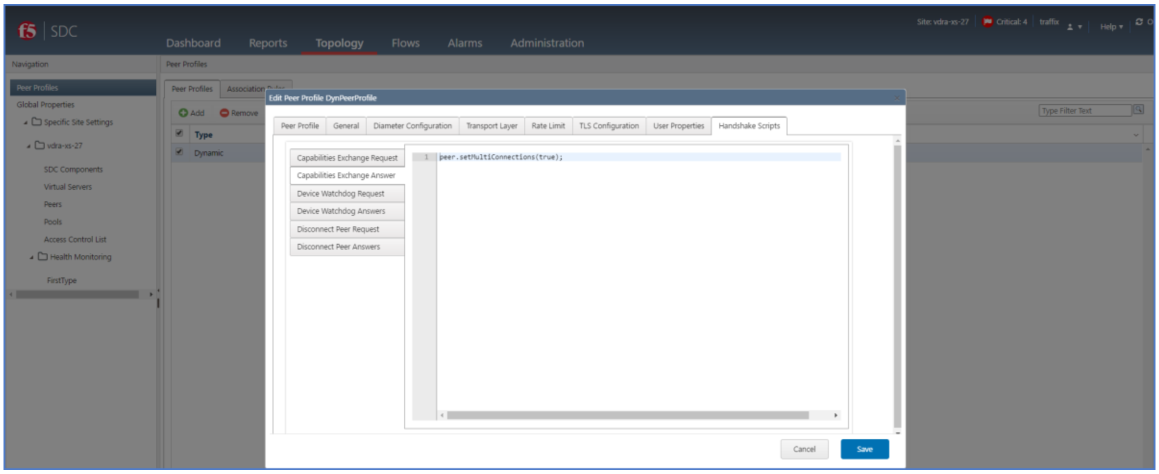
To configure a multi-channel peer connection:

1. Go to **Topology** > **Peer Profiles** > <**the relevant Dynamic Peer Profile**> **Edit**.

2. In the Edit Peer Profile window, select **Handshake Scripts**>**Capabilities Exchange Answer**.

3. Add the following command to the script: peer.setMultiConnections(true).

Figure 6: Multi-Connection Script for Client Peers



```
peer.setMultiConnections(true);
```

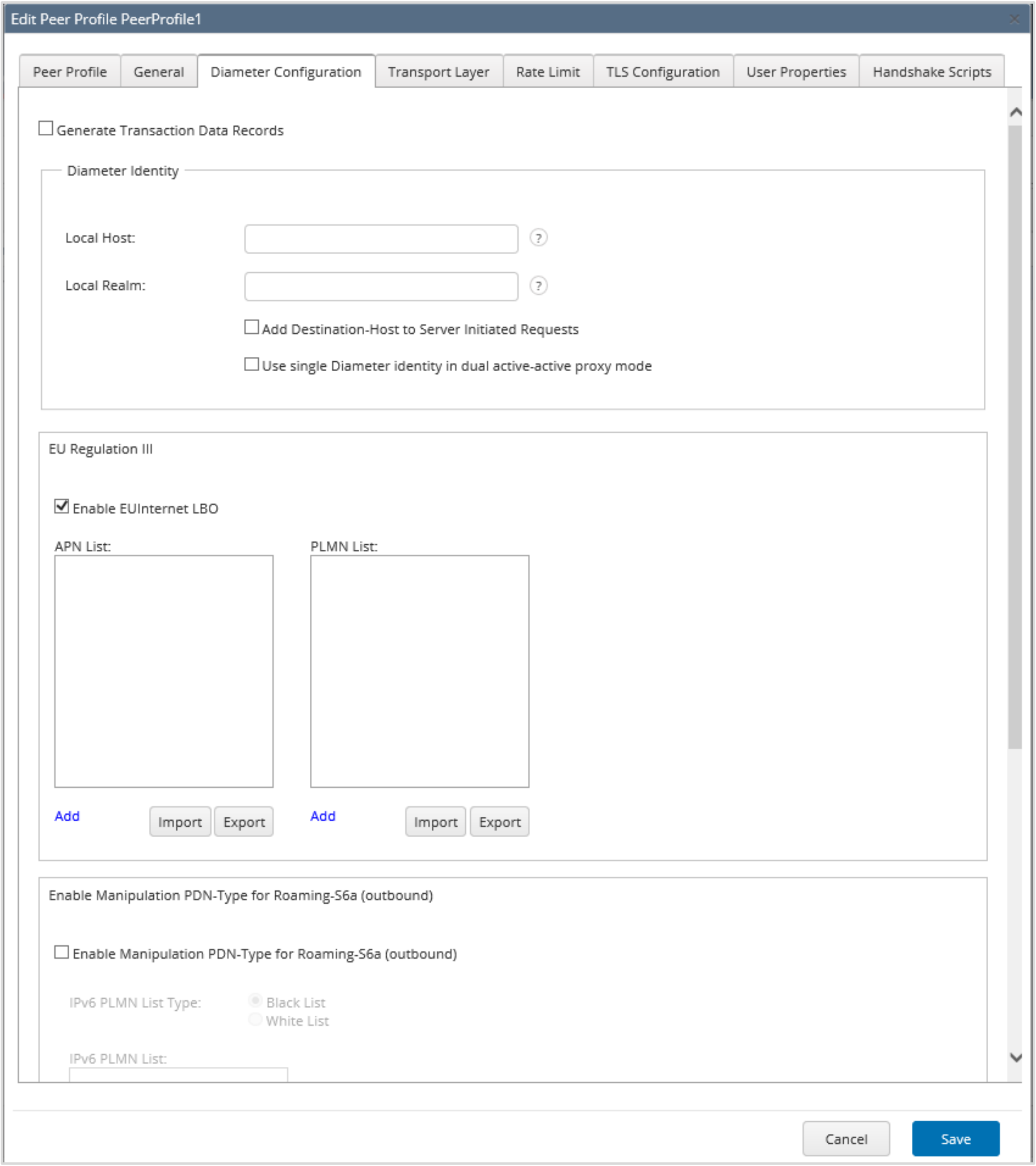
All subsequent peer connections will appear in the **Topology** > **Peers** screen as new peers with the same Origin Host name, followed by a suffix (SN_<#>).

3.3.1.5 Diameter Peer Profile

This section continues with the next wizard steps for adding a Diameter peer profile.

Under the **Diameter Configuration** tab, you can configure the Diameter Identity, EU Regulation LBO Breakout, IPv6 - IPv4 Enablement, Loop Detection, Transaction Data Records (TDRs), and Idle Connection Time.

Figure 7: Diameter Configuration



Edit Peer Profile PeerProfile1

☐ Generate Transaction Data Records

Diameter Identity

Local Host:

Local Realm:

☐ Add Destination-Host to Server Initiated Requests

☐ Use single Diameter identity in dual active-active proxy mode

EU Regulation III

☑ Enable EUInternet LBO

APN List:

PLMN List:

Enable Manipulation PDN-Type for Roaming-S6a (outbound)

☐ Enable Manipulation PDN-Type for Roaming-S6a (outbound)

IPv6 PLMN List Type: ○ Black List ○ White List

IPv6 PLMN List:

- **Generate Transaction Data Records**

By default, the SDC collects and displays information for specific message AVPs. You can configure TDRs to be generated on a peer profile basis.

Note: If you want to add additional AVPs to the default ones that are reflected in the TDRs, define the TDRs according to a routing rule (see Defining TDRs).

To generate TDRs per peer profile:

Select **Generate Transaction Data Records**.

- **Diameter Identity**

1. You can define the values for the message’s origin-host and origin-realm that will override the default values. By default, the message’s origin-host AVP value is the name of the message’s virtual server, and the message’s origin-realm AVP value is configured per FEP and is taken from the FEP that the virtual server is configured to use.

The Diameter identity policy selected when defining the routing rules definition will take the values defined here, and replace the message AVPs according to the selected policy. For more information about the Diameter identity policies, see *Configuring Diameter Identity* .

1. **To define the Diameter Identity values:**

2. In **Local Host**, set the value you wish to appear as the message’s origin-host.

3. In **Local Realm**, set the value you wish to appear as the message’s origin-realm.

4. Select **Add Destination-Host to Server Initiated Requests** to add the Destination-Host, if absent, to server initiated requests when the either the Full or Client Side Proxy policy is selected

Select **Use single Diameter identity in dual active-active proxy mode** to append the Local Host and Local Realm values to requests sent from both FEPs. By default, in active-active mode, the FEP name is also added to the defined values, resulting in two different Diameter identities. This option presents the two FEP instances as a single connection point.

- Note: Prior to selecting this checkbox, the **Local Host** must be defined. As without configuring a **Local Host**, the identity falls back to the default value which is the FEP name, thereby creating different identities for messages that are sent from different FEPs.

- **EU Regulation III Local Breakout for Diameter**

The EU regulation III for Local Breakout facilitates lower cost data roaming for EU mobile users. SDC Diameter peer profiles can be configured with a list of recognized APNs and PLMNs that support EU Local Breakout (LBO). When enabled, the SDC's Local Breakout feature compares the APN of a received ULA/IDR message against the list of supported APNs, and if it matches, continues to check whether the visited network (the message's Origin-Realm (in the case of an ULR) or Destination-Realm (in the case of an IDR)) is in the list of supported PLMNs. There are three possible outcomes:

- It is confirmed that the ULA/IDR message's APN and Origin-Realm of ULR or Destination-Realm of IDR is supported in the APN and PLMN Lists, respectively, the VPLMN-Dynamic-Address-Allowed AVP is changed to true, enabling roaming traffic to be handled locally by the VPLMN (Local Breakout) instead of by the roamer's home network.

- It is confirmed that the ULA/IDR message's APN and Origin-Realm of ULR or Destination-Realm of IDR is supported in the APN, but not in the PLMN Lists, respectively, and then the VPLMN-Dynamic-Address-Allowed AVP is changed to false, thereby preventing roaming traffic to be handled locally by the VPLMN (Local Breakout) and instead is handled by the roamer's home network.

- It is confirmed that the ULA/IDR message's APN and Origin-Realm of ULR or Destination-Realm of IDR is not supported in the APN, and then the PLMN list is not checked, and no changes are made to the VPLMN-Dynamic-Address-Allowed AVP.

To enable and configure EU Local Breakout:

1. Select **Enable EUInternet LBO**.

2. In the **APN List** and **PLMN List** sections, use the **Add**, **Import**, and **Export** options to configure the list to reflect those APNs and PLMNs that are supported by the SDC.

Note: You can only import a CSV file.

- **IPv6 - IPv4 Enablement for Diameter Peer Profiles**

SDC enables modification of the PDN-Type AVP to accommodate for PLMNs that do not support the IPv4v6 mode, to provide operators with greater network flexibility. When enabled, the SDC compares the origin-realm of a Diameter request against the PLMNs included in the PLMN List. The PLMN List can be configured as a Black List, meaning, the origin-realm is compared against all PLMNs not listed in the PLMN List or as a White List, meaning the origin-realm is compared against only those PLMNs included in the PLMN List. If it matches, the SDC modifies the PDN-Type parameter from “2” (IPv4v6) to “0” (IPv4) for PLMNs that do not support IPv4v6 mode.

1. **To enable IPv6 protocol for roaming:**

2. Select **Enable Manipulation PDN-Type for Roaming-S6a (outbound)** for Diameter peer profiles.

3. Select the **Black/White List** radio button depending on if you want to exclude or include, respectively, those PLMNs that are listed in the PLMN List not to be transformed to IPv6.

In the IPv6 **PLMN List** section use the **Add**, **Import**, and **Export** options to configure the list to reflect those PLMNs that are supported by the SDC.

Note: You can only import a CSV file.

Click **Save**.

To configure a watchdog timer:

4. In **Idle Connection Time –TW (ms)**, set the time (in milliseconds) that the connection will remain open, without any traffic, before sending a watchdog request. 1.

Note: The default value is 30,000 ms. The server will not be reconnected until traffic resumes. This parameter is only relevant in keep alive mode.

- Transport Layer Options

Transport Layer Options can be configured by peer profile or globally (**Administration>Transport Layer**). The default is to use the Administration Configuration (**Add New Peer Profile>Transport Layer>Use Administration Configuration**). By selecting **Use Administration Configuration** for a peer profile, you are configuring the system to apply the defined parameters that were defined in **Administration>Transport Layer** for a site in which the relevant peers match the peer profile.

1. **To configure the Transport Layer Options per peer profile:**

2. Under the **Transport Layer Options** tab, select **Override Administration configuration**.

Set the parameters that control the behavior of transport layer channels. For information on the transport layer options, see Default Transport Configuration.

1.

- Rate Limits

2. **To configure the rate limit:**

Under the **Rate Limit** tab:

Set the thresholds of the data flow, which prevent data from overloading the system. For information on Rate Limits, see *Configuring Rate Limits*.

- TLS Configuration

To configure the TLS Configuration:

Under the **TLS Configuration** tab, select one of the following:

- **No TLS Security**

1. - **Pre Capabilities Exchange TLS**

- **Post Capabilities Exchange TLS**

Note: In the Post Capabilities Exchange TLS, the TLS handshake begins when the client and server are both in open state, after completion of the CER/CEA exchange. If the handshake is successful, all further messages are sent via TLS.

In the Pre Capabilities Exchange TLS, the TLS handshake begins prior to any Diameter message exchange. All Diameter message are sent through the TLS connection after a successful setup.

---

2. If you select either **Pre** or **Post Capabilities Exchange TLS**, you have the option to change the default **TLS Keystore Password** and **TLS Trust Store Password**. Default passwords are generated as part of the automatic TLS security key generation. The TLS security key secures the connections between the SDC and its

3. connected peers.

Click **Add Cipher Suite** to add a TLS cipher suite.

Note: Cipher Suite changes in Peer Profiles only takes effect after SDC processes are restarted.

4. Cipher suites represent the combined names of various activities which are performed during the negotiation on security settings for network connection.

Click **Save**.

- Handshake Scripts

The Peer Profile Handshake page contains tabs for the six scripts that can be configured. Each script corresponds to one of the connectivity related messages (requests and answers with a dedicated script each) that are sent between the SDC and the peer – messages sent to establish the connection (the capabilities exchange request and answer scripts), maintain the connection (the device watchdog request and answer scripts), and terminate the connection (the disconnect peer request and answer scripts).

Figure 8: Peer Profile Handshake Screen

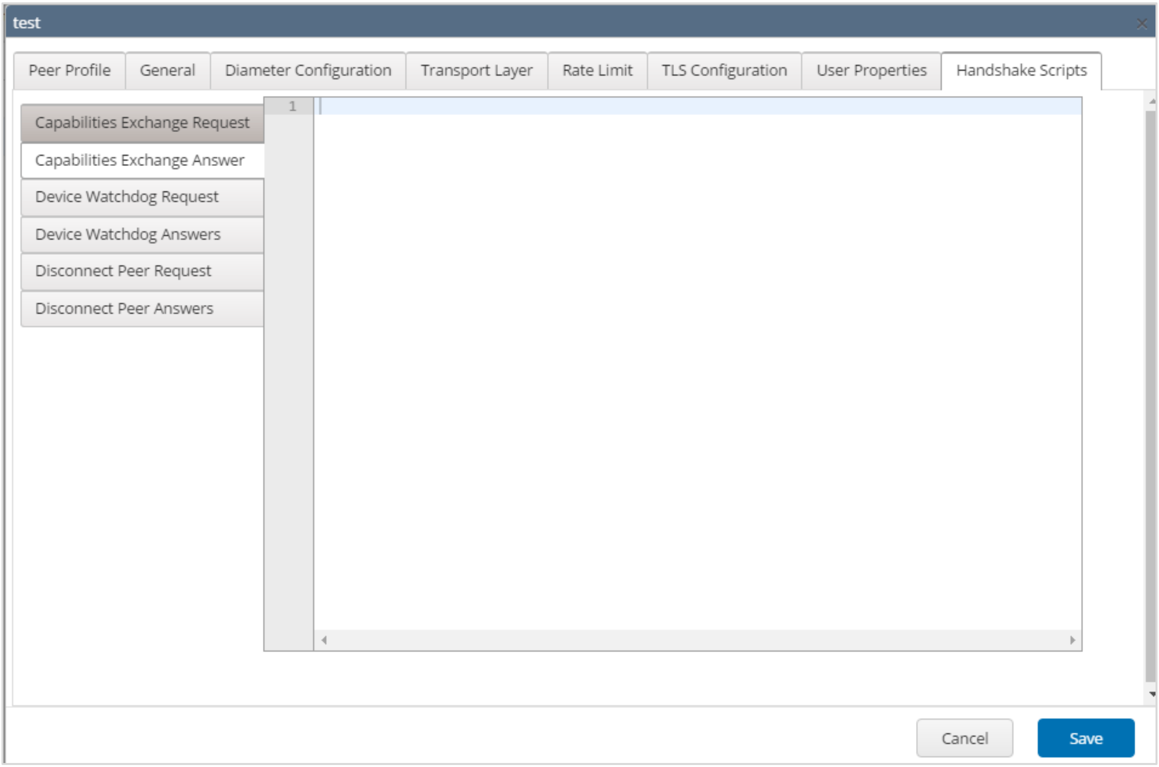


Table 9 details the parameters SDC provides to the scripts:

Table 8: Request and/or Answer Scripts Parameters

Parameter	Type
Request	Message
Peer	Peer
Stack	Stack

Parameter	Type
userTraceLogger	UserTraceLoggerWrapper
Metadata	MetaData

3.3.1.5.1 Configuring Diameter Protocol Loop Detection

Loop Detection avoids endless message loops, by preventing the same Diameter message being sent to the same network element more than once. A loop can be detected locally by the SDC or by a remote node:

Local Loop Detection: each time a message is routed to an SDC, the SDC's name is registered to the Route-Record AVP. If a message is routed through the SDC twice, the SDC recognizes its own identity in the Route-Record AVP and a loop is detected. When a loop is detected, the SDC stops handling the message and the message is discarded. Loop detection by the SDC is configured by peer profile.

Note: In a geo-redundant deployment, the default Local Loop Detection configuration is to check the Route-Record AVP on each site. You may want to override this configuration either globally or partially, depending on if the deployment is configured with the same routing domain (DRA/DEA) or a mix-domain configuration.

To override the geo-redundant Local Loop Detection globally, go to **Topology** > **Global Properties** and add the **Globalsdccld** parameter and change the **Value** to something other than the default value of F5_SDC (see *Global Properties*). To override geo-redundant loop detection per site, reconfigure the site's **Globalsdccld** parameter's value (see *Configuring a Site's User Properties*).

- Remote Node Loop Detection: a remote node detects a loop (based on the AVP identifiers) and sends an error message to the SDC. Remote Node Loop Detection is configured in routing rule scripts. For remote node loop detection, you can also configure how to handle the detected message.

To enable SDC local loop detection:

Go to **Topology** > **Peer Profiles** > **Add Peer** > **Diameter Configuration**.

1. Note: The default is **Disable Loop Detection**. When selected, the SDC will not detect any loops for the peer profile.

Select **Loop Detection by Source**.

2. **To configure remote node loop detection:**

1. Go to **Flows** > **Flows** > **<Default>** > **Routing Rules**.

2. Select a routing row.

3. Under the **Handle Errors** tab:

4. Select **Check Error in Answer** and add in a script to enable remote node loop detection with the tag "REMOTE_LOOP_DETECTED".

The following is an example of a script:

```
//def resultCode = answer.getValue("Result-Code");
//if (resultCode == 3005) { //Loop detected
//            return RemoteNodeEvent.REMOTE_LOOP_DETECTED;
//}
```

5. Select **Handle Server/Client Error** and add in a script to configure how a detected loop message should be handled:

The following is an example of a **Handle Client Error** script:

```
if (event == RemoteNodeEvent.LOCAL_LOOP_DETECTED) {
  if (protocol == Protocol.Diameter) {
    def answer = requestFromClient.createAnswer(3005);//Loop detected
    return answer;
```

- Statistics and Logging

The following statistics are collected and stored in the EMS site:

- The number of local loop detection events per SDC ("Peer Local Loop events").
- The number of remote loop detection events per Remote Peer ("Peer Remote Loop events".

An Info log message is generated when a loop is detected ("Local loop by source is detected. SDC global identifier: {0}, is contained in the incoming request. At transaction event: {1}").

3.3.1.6 HTTP Peer Profile

This section continues with the next wizard steps for adding an HTTP peer profile.

To configure an HTTP peer profile:

1. Under the **HTTP Configuration** tab, in **Max Connection Count Limit (Per Client)**, set the maximum number of open HTTP connections.

2. In **Idle Connection Time – TW (ms)**, set the time (in milliseconds) that the connection will remain open, without any traffic.

Note: The default value is 30,000 ms. The server will not be reconnected until traffic resumes. This parameter is only relevant when in keep alive mode.

3.

Note: The default value is 30,000 ms.

1.

2.

To configure the Transport Layer Options:

Under the **Transport Layer Options** tab:

1. Set the parameters that control the behavior of transport layer channels. For information on the transport layer options, see *Replicating Session Data*.

To configure the rate limit:

Under the **Rate Limit** tab:

Set the thresholds of the data flow, which prevent data from overloading the system. For information on Rate Limits, see *Configuring Rate Limits*.

To configure the TLS Configuration:

2. Under the **TLS Configuration** tab, select one of the following:

- **No TLS Security**
- 1. **Pre Capabilities Exchange TLS**
- **Post Capabilities Exchange TLS**

Note: In the Post Capabilities Exchange TLS, the TLS handshake begins when the client and server are both in open state, after completion of the CER/CEA exchange. If the handshake is successful, all further messages are sent via TLS.

In the Pre Capabilities Exchange TLS, the TLS handshake begins prior to any Diameter message exchange. All Diameter message are sent through the TLS connection after a successful setup.

2. If you select either **Pre** or **Post Capabilities Exchange TLS**, you have the option to change the default **TLS Keystore Password** and **TLS Trust Store Password**. Default passwords are generated as part of the automatic TLS security key generation. The TLS security key secures the connections between the SDC and its connected peers. 3.

Under **Cipher Suite,** add a TLS cipher suite.

Note: Cipher Suite changes in Peer Profiles only takes effect after SDC processes are restarted. 4.

Cipher suites represent the combined names of various activities which are performed during the negotiation on security settings for network connection.

Click **Save**. The Peer Profile Handshake page is displayed.

3.3.1.7 LDAP Peer Profile

This section continues with the next wizard steps for adding an LDAP peer profile.

To configure the rate limit:

1. Under the **Rate Limit** tab, set the thresholds of the data flow, which prevent data from overloading the system. For information on Rate Limits, see *Configuring Rate Limits*.

2. Click **Save**.

3.3.1.8 RADIUS Peer Profile

This section continues with the next wizard steps for adding a RADIUS peer profile.

To configure the authorization (COA) and authentication attributes:

1. Under the **RADIUS Configuration** tab:

    a. In **COA Listening Port**, set the listening port that you want to define as the authorization port.

    b. Select **Use Message-Authenticator** if you want to use the authenticate message feature and then select the algorithm to be used to authenticate RADIUS messages from the drop-down.

    Note: Messages containing the “EAP-Message” attribute is authenticated automatically using a default algorithm (HmacMD5), therefore there is no need to configure this field.

    c. Select **Validate Message-Authenticator**, if you want to validate each RADIUS message.

    Note: Messages containing the “EAP-Message” attribute are authenticated automatically using a default algorithm (HmacMD5), therefore there is no need to configure this field.

To configure the UDP options:

1. Under the **UDP Options** tab:

   a. In **Duplicate Request Answer Persistence Timeout**, set the time frame in which to persist the returned answer, in order to answer further duplicated requests.

   b. In **Duplicate Request Pending Answer**, set the time frame in which to wait for the answer to be returned and for discard further duplicated requests.

   c. In **Duplicate Request Handling Policy**, select whether to resend (the previously cached response) or discard duplicated messages.

To configure the rate limit:

1. Under the **Rate Limit** tab:

2. Set the thresholds of the data flow, which prevent data from overloading the system. For information on Rate Limits, see *Configuring Rate Limits.*

3.3.1.9 SS7 Peer Profile

This section continues with the next wizard steps for adding an SS7 peer profile.

- **EU Regulation III Local Breakout for SS7 Peer Profiles**

  The EU regulation III for Local Breakout, facilitates lower cost data roaming for EU mobile users. SDC SS7 peer profiles can be configured with a list of recognized APNs and PLMNs that support EU Local Breakout (LBO). When enabled, the SDC's Local Breakout feature compares the APN of a received InsertSubscriberData request against the list of supported APNs, and if it matches, continues to check if the request's SCCP Called Party Address is in the list of supported PLMNs. Once it is confirmed that the request's APN and PLMN are supported, a vplmnAddressAllowed parameter is added to the request, enabling a connection (Local Breakout) to be established with a VPLMN.

Note: The IsSccpMode attribute in the site topology file must be defined with a “true” value during installation to enable this feature for SS7 configured peer profiles.

To enable and configure EU Local Breakout:

1. Under the **MAP Manipulations** tab, select **Enable EUInternet LBO**.
2. In the **APN List** and **PLMN List** sections, use the **Add**, **Remove**, **Import**, and **Export** options to configure the list to reflect those APNs and PLMNs that are supported by the SDC.

Note: You can only import a CSV file.

- **IPv6 - IPv4 Enablement for SS7 Peer Profiles**

SDC enables modification of the Ext-PDP-type parameter to accommodate for PLMNs that do not support the IPv4v6 mode, to provide operators with greater network flexibility. When enabled, the SDC compares the SCCP address of an SS7 request against the PLMNS included in the PLMN List, and if it matches, the Ext-PDP-type parameter is removed for PLMNs that do not support IPv4v6 mode. The PLMN List can be configured as a Black List, meaning, the SCCP address is compared against all PLMNs not listed in the PLMN List or as a White List, meaning the SCCP address is compared against only those PLMNs included in the PLMN List.

Note: The IsSccpMode attribute in the site topology file must be defined with a “true” value during installation to enable this feature for SS7 configured peer profiles.

To enable IPv6 protocol for roaming:

1. Select **Enable Manipulation Ext-PDP-Type for Roaming-Gr (outbound)**.
2. Select the **Black/White List** radio button depending on if you want to exclude or include, respectively, those PLMNs that are listed in the PLMN List not to be transformed to IPv6.

In the **IPv6 PLMN List** section use the **Add**, **Remove**, **Import**, and **Export** options to configure the list to reflect those PLMNs that are supported by the SDC.

3. Note: You can only import a CSV file.

Click **Save**.

3.3.2 Configuring a Site's User Properties

4.

You can either configure user properties per site or per a peer or pool that is part of a site. When user properties are configured per peer or pool, the SDC invokes those values prior to user property values that are configured per site.

To configure user properties for a site:

1. Go to **Topology** > **Site** > **Add**.
2. In the **Name** field, enter a user friendly property name.
3. In the **Value** field, enter the desired value for the property.
4. In the **Path** field, the path name for the property is displayed.

Note: For example, you can configure an Origin Host and Origin Realm for a site instead of the **Local Host/Local Realm** of a remote peer. To do so, under the **Name** field, type in "site-origin-host" and "site-origin-realm."

3.3.3 Configuring the SDC Components

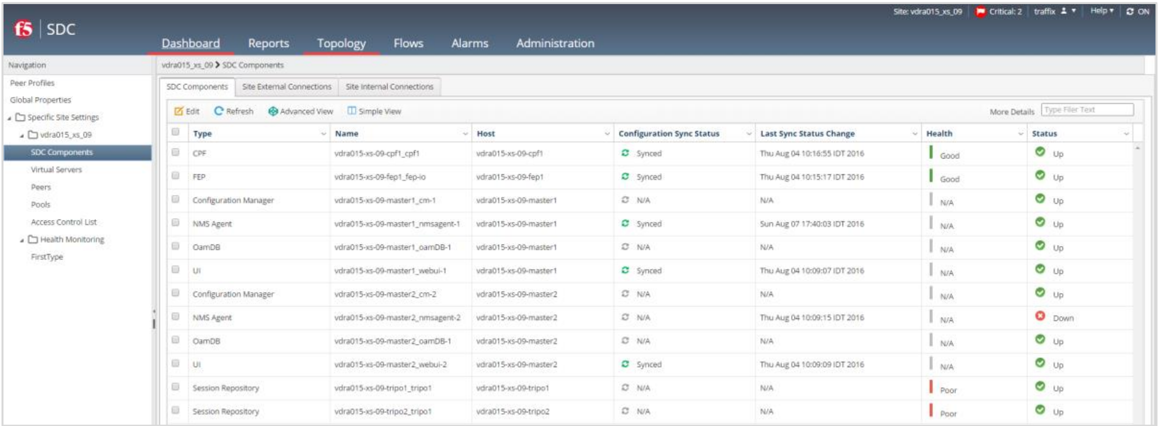
This section describes how to view and edit the SDC Components that were defined for
1. the SDC site during the installation process.

To edit the SDC Component list:

Go to **Topology** > **Specific Site Settings** > **Site** > **SDC Components> SDC Components**

The list of SDC Components defined throughout the installation procedure is displayed according to the properties described in *Table 10*.

Figure 9: SDC Components



Type	Name	Host	Configuration Sync Status	Last Sync Status Change	Health	Status
CPF	vdra015-xs-09-cpf1_cpf1	vdra015-xs-09-cpf1	Synced	Thu Aug 04 10:16:55 IDT 2016	Good	Up
FEP	vdra015-xs-09-fep1_fep-io	vdra015-xs-09-fep1	Synced	Thu Aug 04 10:15:17 IDT 2016	Good	Up
Configuration Manager	vdra015-xs-09-master1_cm-1	vdra015-xs-09-master1	N/A	N/A	N/A	Up
NMS Agent	vdra015-xs-09-master1_nmsagent-1	vdra015-xs-09-master1	Synced	Sun Aug 07 17:40:03 IDT 2016	N/A	Up
OamDB	vdra015-xs-09-master1_oamDB-1	vdra015-xs-09-master1	N/A	N/A	N/A	Up
UI	vdra015-xs-09-master1_webui-1	vdra015-xs-09-master1	Synced	Thu Aug 04 10:09:07 IDT 2016	N/A	Up
Configuration Manager	vdra015-xs-09-master2_cm-2	vdra015-xs-09-master2	N/A	N/A	N/A	Up
NMS Agent	vdra015-xs-09-master2_nmsagent-2	vdra015-xs-09-master2	N/A	Thu Aug 04 10:09:15 IDT 2016	N/A	Down
OamDB	vdra015-xs-09-master2_oamDB-1	vdra015-xs-09-master2	N/A	N/A	N/A	Up
UI	vdra015-xs-09-master2_webui-2	vdra015-xs-09-master2	Synced	Thu Aug 04 10:09:09 IDT 2016	N/A	Up
Session Repository	vdra015-xs-09-tripo1_tripo1	vdra015-xs-09-tripo1	N/A	N/A	Poor	Up
Session Repository	vdra015-xs-09-tripo2_tripo1	vdra015-xs-09-tripo2	N/A	N/A	Poor	Up

Table 9: SDC Components

Column	Description
Type	The role that the component was configured to fill. Each component is defined by a type, based on the role that it fulfills in the installed site.
Name	The name of the component, as defined in the site topology file.
Host	The name of the site machine that the component runs on.
Configuration Sync Status	Indicates if the component's configuration data is currently synced with the configuration manager's data.
Last Sync Status Change	Indicates the most recent date and time that the component's configuration data was synchronized with the configuration manager's data. This includes synchronization when the connection between the component and a configuration manager has been reestablished, after the connection was temporarily lost between the component and both configuration managers.
Health	Indicates the health status of the CPF. FEP, and Session Repository. The health of theses component is based on a compilation of health statistics. Note: You can click **More details** to view the health-related statistics for all the components.

Column	Description
	The component health is presented with one of three possible states: 🟩 Good 🟨 Fair 🟥 Poor The component health state is determined by the worse state of at least one statistic/parameter. For example, if at least one of the parameters is red, the health will be red. If there are only green and yellow indications and at least one of them is yellow, the health will be yellow. If all the parameters are green, the health parameter will be green. The collected statistics can be viewed in the bottom pane by selecting one of these components. The health state is based on the following statistics (per last minute) which are displayed for a selected component in the **Health Parameters** tab. There are three display options: ✅ – no threshold was reached, ⚠ – a minor or major threshold was reached, ❌ – a critical threshold was reached. Note: As the ⚠ icon relates to both minor and major thresholds, the color box surrounding the actual statistical number can be yellow or orange, depending if a minor or major threshold was reached. For CPF Components: ▪ CPF Errors ▪ Incoming Request Queue usage ▪ Incoming Answers Queue usage ▪ Session Repository Incoming Request Queue Usage ▪ Diameter Pending Requests Queue Usage ▪ Status: ▪ Active Alarms For FEP Components: ▪ FEP Errors

Column	Description
	▪ Status: ▪ Active Alarms For Session Repository Components: ▪ Session Repository Storage Usage ▪ Session Repository Message Queue ▪ Session Repository Mate ▪ Status: ▪ Active Alarms Additional Information ▪ Full Repository Replication: ▪ SRR Repository Replication:
Status	Indicates the component's status (Up. Down).
Alarm	Indicates – by color – the highest severity alarm currently raised for this component, and is a link to view the active alarms.
Collected Statistics	
SDC Error Ratio	The percentage of transactions processed by the component that returned an error response, out of the total processed transactions by the component.
SDC Errors	The number of transactions processed by the component that returned an error response
SDC Received Requests	The number of total requests processed by the component
Incoming Request Queue Usage (messages)	The number of request messages currently in the component's incoming queue.
Incoming Answers	The number of answer messages currently in the component's incoming queue.

Column	Description
Queue Usage (messages)	
Diameter Pending Requests Queue Usage	The percentage of the Diameter pending request queue that is in use.
Session Repository Incoming Request Queue Usage	The percentage of the Session Repository incoming request queue that is in use.
Session Repository Storage Usage	The percentage of used Session Repository storage.
Session Repository Message Queue	The state of the Session Repository (SRR) queue.
Session Repository Mate	Indicates if the paired Session Repository instance is up and connected.
Full Repository Replication	Indicates the status of a full replication between two session repositories.
SRR Repository Replication	Indicates the status of a replication between two session repository SRR queues.

1.

To edit the properties for a FEP or CPF component:

Select the row of the FEP or CPF component and click **Edit**.

The SDC Component Properties window displays the following properties:

- **General**
- **Diamete**r
- **SS7**
- **User Properties**

After editing the parameter properties, click **Save**.

2. The following tables describe the parameter for of these properties.

Table 10: General SDC Component's Properties

Parameter	Description
URI	Universal Resource Identifier. Describes the identity of the SDC Component. Used during capability exchange and routing. Cannot be modified. e.g. aaa://SDC Note: The URI is provided during SDC's installation procedure. For more information on the installation procedure, see the relevant installation documentation.
Product Name	The product name of the SDC Component, published during capability exchange.
Reestablish Connection Time (TC)	The interval for reconnecting the SDC component. Note: The value must be between 1-30000 milliseconds.

Table 11: Diameter SDC Component's Properties

Parameter	Description
Idle Connection Time (TW)	Watchdog and reconnection timer (in milliseconds). e.g. 30000. Note: The minimum TW value is 6000 milliseconds.

Parameter	Description
Supported Application IDs	Defines the supported Diameter applications (comma separated), and hence defines the Diameter messages that the SDC Component may handle. e.g. Ro, Gx. Note: For a full list of the supported applications, see *Appendix B: Supported Application Identifiers*. You can add additional application IDs in the standard_dynamic_example.txt file (located in the /opt/traffix/sdc/config folder) or using the Web Service API Method: SetDiameterPropertiesforNode, in thesupportedApplicationIds parameter. For more information, see the Troubleshooting Guide.
Supported Vendor IDs	Supported Vendor IDs that the SDC declares and sends as part of Capability Exchange.
Vendor Id	Used as the published Vendor ID during capability exchange e.g. 27611.
Routing Resend Tries	The maximum resend attempts.
Routing Resend Wait Time	The time interval between two resends attempts.
Realm	The Diameter realm to which SDC belongs. Used during capability exchange, e.g. F5.com

Table 12: SS7 SDC Component's Properties

Note: The SS7 protocol is only supported in bare-metal deployments.

Parameter	Description
SS7 Hlr Number	This parameter currently not supported.
SS7 Component Value Max Size	The maximum message size for insertSubscriberDataArg SS7 messages that were converted from Diameter ULAs.
Point Code	The local point code.

Table 13: SDC Component's User Properties

Parameter	Description
Name	Enter a user friendly property name
Value	Enter the desired value for the property
Path	The path name for the property is displayed

Note: User properties can also be defined using the setEntityProperties Web Service API method and retrieved using the getEntityProperties Web Service API method or using Groovy scripting. For more information about the Web Service API methods, see *F5 SDC Web Services API Guide*.

To refresh the SDC Component list:

1. Click **Refresh**.

Note: Each IP Address used by the SDC Component should be separately licensed in order for it to operate. For more information on SDC's licensing mechanism, see *Licensing the FEPs*.

3.3.3.1 Viewing the External Connections

An SDC site may be connected to an EMS site and to additional SDC sites. The Site External Connections screens provides an overview of these connections and their status.

To view the site external connections:

1. Go to **Topology** > **Specific Site Settings** > **Site** > **SDC Components** > **Site External Connections**

Table 15 describes the information provided for these connections.

Table 14: Site External Connections

Column	Description
Connected Site	The type of site that the monitored site is connected to.
Component	The type of site component that the monitored site is connected to.
Component Name	The name of the site component that the monitored site is connected to.
Last Status Time	The date and time that the connection status was queried.
Status	The connection status (Connected, Disconnected) between the sites.

3.3.3.2 🌐 Viewing the EMS – SDC Site Connections

From an EMS Web UI, you can view the connectivity status between the EMS configuration manager and the SDC site configuration manager.

To view the connectivity status:

1. Go to **Topology** > **SDC Sites**.

Table 16 describes the provided information. -

Table 15: EMS-SDC Site Connectivity

Column	Description
Site Name	The name of the monitored site that is connected to the EMS.
Configuration Manager Connectivity Last Status Change	The date and time that the connection status was queried.
Configuration Manager Connectivity Last Status	The connection status (Connected, Disconnected) between the sites.

2. Select a site to view the site’s SDC component information in a bottom table pane.

3.3.3.3 Viewing the Internal Connections

All components installed on an SDC site are connected to and communicate with other components within the site. The Site Internal Connections screens provides an overview of these connections and their status.

To view the site internal connections:

Go to **Topology** > **Specific Site Settings** > **Site** > **SDC Components** > **Site Internal Connections**

1.

Table 17 describes the information provided for these connections.

Table 16: Site Internal Connections

Column	Description
Component Name	The name of the monitored component.
IP Address	The IP address of the monitored component.
Host	The site machine that the monitored component runs on.
Connected Component	The name of the connected component.
Connected Component IP Address	The IP address of the connected IP and port.
Connected Component Host	The site machine that the connected component runs on.
Status	The connection status (Connected, Disconnected) between the components.

3.3.4 Configuring Virtual Servers

Virtual Servers are virtual instances of SDC used to facilitate every protocol used by SDC to communicate with the Remote Clients. You should create a single Virtual Server per each protocol that SDC listens to in your network. This section describes how to view and add the different virtual servers.

3.3.4.1 Viewing the Virtual Servers

You can view a list of Virtual Servers that were defined during the installation process.

To view a current list of Virtual Servers:

Go to **Topology** > **Specific Site Settings** > **Virtual Servers**.

1. The list of Virtual Servers is displayed according to the properties described in *Table 18*.

Table 17: Virtual Server Properties

Column	Description
Name	A user-friendly display name assigned to the Virtual Server. e.g. VS1
FEP Group	The FEP Node through which the Virtual Server connects to SDC. The virtual server's configuration is used by the designated FEP.
Protocol	The signaling protocol/s used by the Virtual Server. e.g. Diameter
Peer Profile	The associated Peer Profile
Administrative State	Indicates whether the Virtual Server is connected (enabled) to SDC or disconnected (disabled) from it
Status	Indicates if the Virtual Server is **Open** (✓) or **Closed** (✖) or **Limited** (❗) to receive traffic.
FEP Node Connected to a Selected Virtual Server	
Node Name	The name of the FEP that is connected to a selected virtual server
Address	The address
Status	Indicates the status (**Open, Closed**, and **Not Available**) for the FEP

1.

To edit a Virtual Server Property:

Select a Virtual Server and then select **Edit**, **Enable** or **Disable**.

Note: Prior to changing a virtual server’s peer profile, you must disable and then enable the virtual server for it to recognize the new peer profile.

3.3.4.2 Adding a New Virtual Server

You can add a virtual server in addition to those that were configured during the installation process.

To add a new Virtual Server:

1. Go to **Topology** > **Specific Site Settings** > **Site** > **Virtual Servers** >**Add**. The Add Virtual Server window appears.

2. In the **Name** field, enter a user friendly display name to identify the Virtual Server. e.g. VS1.

Note: When implementing the Diameter Identity mechanism, this value is used as the default value for the message’s origin-host AVP.s

Warning: After submitting the new Virtual Server, its name may not be modified.

1.

2. In the **Port** field, enter the port on which the virtual server is listening.

In the **Protocol** field, from the drop-down list, select the protocol used by the Virtual Server (for example, **Diameter**, **RADIUS**, **HTTP**, **LDAP**).

The wizard configuration options display according to your protocol selection. Proceed to the next section according to your protocol selection.

Note: The timeout after which SDC disconnects the channel through the virtual server (if no messages are passed on it) is determined by the .xml configuration file parameter **TCPIdleTimer** (which has a default value of ten seconds).

Note: The added virtual server has a **Closed** (✖) status by default, until it has been assigned a license. An alarm (sdcVirtualServerStateChanged) is generated, indicating that the virtual server is missing a license and is unable to process traffic.

For information about adding a license, see *Licensing the FEPs*.

3.3.4.2.1 Diameter Virtual Server

This section continues with the Add Virtual Server wizard for adding a Diameter virtual server.

To add a Diameter virtual server:

1. In **FEP Group**, select the Proxy Group on which the virtual server is set.
2. In **Peer Profile**, from the drop-down, select the Peer Profile associated with this Virtual Server.
3. Select **Use SCTP Transport** to use SCTP when in message transport (rather than TCP).
4. Select **Use for Geo Redundant Sites Connection** to enable the peer to handle proxied requests.
5. Click **Save**. The new Diameter Virtual Server is displayed in the Virtual Server table.

3.3.4.2.2 RADIUS Virtual Server

This section continues with the Add Virtual Server wizard for adding a RADIUS virtual server.

To add a RADIUS virtual server:

1. In **FEP Group**, select the Proxy Node (FEP Node) on which the virtual server is set.
2. In **Peer Profile**, from the drop-down, select the Peer Profile associated with this Virtual Server.

3. Click **Save**. The new RADIUS Virtual Server is displayed in the Virtual Server table.

3.3.4.2.3 HTTP Virtual Server

This section continues with the Add Virtual Server wizard for adding an HTTP virtual server.

To add an HTTP virtual server:

1. In **FEP Group**, from the drop-down, select the Proxy Node (FEP Node) on which the virtual server is set.
2. In **Peer Profile**, from the drop-down select the Peer Profile associated with this Virtual Server.
3. Select **Close Connection on Answer** to close the connection with the Remote Client/Server upon Answer retrieval.
4. Select **Use for Geo Redundant Sites Connection** to enable the peer to handle proxied requests.
5. Click **Save**. The new HTTP Virtual Server is displayed in the Virtual Server table.

3.3.4.2.4 LDAP Virtual Server

This section continues with the Add Virtual Server wizard for adding an LDAP virtual server.

To add an LDAP virtual server:

1. In **Num Acceptor Threads**, set number of threads to be used.
2. In **Back Log**, type in the queue size for incoming LDAP messages waiting to be handled by the LDAP virtual server.
3. Select **Bind User** to mandate user credentials.
   a. In **Bind User**, type in the LDAP user for the directory server authentication.

In **Bind Password**, type in the LDAP user's password for the directory server authentication.

b. Select **Anonymous Bind** to allow users to connect to the directory without user credentials.

4. Click **Save**. The new LDAP Virtual Server is displayed in the Virtual Server list.

5.

3.3.5 Configuring Peers

This section describes how to configure the different peers.

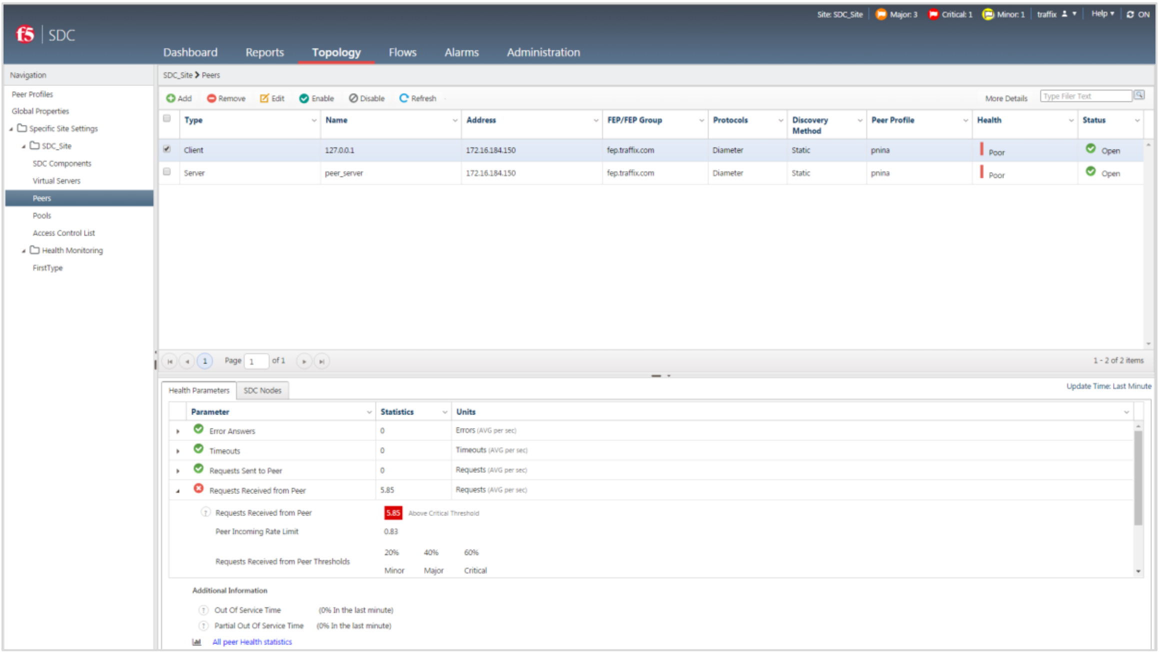
3.3.5.1 Viewing the List of Peers

To view the current list of Peers:

1. Go to **Topology** > **Specific Site Settings** > <**Site Name**>> **Peers**.

   The list of currently defined peers is displayed.

Figure 10: Peers



☐	Type	Name	Address	FEP/FEP Group	Protocols	Discovery Method	Peer Profile	Health	Status
☑	Client	127.0.0.1	172.16.184.150	fep.traffix.com	Diameter	Static	pnina	Poor	✓ Open
☐	Server	peer_server	172.16.184.150	fep.traffix.com	Diameter	Static	pnina	Poor	✓ Open

Parameter	Statistics	Units
✓ Error Answers	0	Errors (AVG per sec)
✓ Timeouts	0	Timeouts (AVG per sec)
✓ Requests Sent to Peer	0	Requests (AVG per sec)
✗ Requests Received from Peer	5.85	Requests (AVG per sec)
Requests Received from Peer	5.85 Above Critical Threshold	
Peer Incoming Rate Limit	0.83	
Requests Received from Peer Thresholds	20% Minor, 40% Major, 60% Critical	

Additional Information

Out Of Service Time (0% In the last minute)

Partial Out Of Service Time (0% In the last minute)

Table 19 presents a list of peers.

Table 18: Peer's Properties

Column	Description
Type	Indicates if the peer is a client or server
Name	A user-friendly unique display name assigned to the Remote Peer. e.g. Server 1
Addresses	The address (single or multiple) of the Remote Peer (client or server) and the port number used by it to access the SDC Components, to send and receive protocol messages. e.g. 1.1.1.1
FEP/FEP Group	The name of the FEP Node or FEP Group to which the peer is connected
Protocols	The signaling protocol/s used by the client or server. e.g. Diameter/JMS
Discovery Method	Specifies whether the peer was statically configured, or dynamically discovered. Server Peers must be statically configured. Client Peers may be dynamically discovered, or, in case the SDC Component does not allow unknown peers to connect, must be statically configured too (For more information, see *Configuring the Access Control List*). Note: Traditionally, Remote Clients are dynamically discovered by SDC. Static Discovery method is used in case one wishes to limit the number of Remote Clients and defines specific Remote Clients in the system.
Peer Profile	Peer Profile is an attribute that may be assigned to the peer. Peers assigned with a Peer Profile are handled in a predefined manner – they may receive unique messages and send unique messages.
Health	Indicates the health status of each peer. The health of the peer is based on a compilation of health statistics. Note: You can click **More Details** and scroll to the right to view the health related statistics for all peers. The peer health is presented with one of three possible states: 🟩 Good 🟨 Fair

Column	Description
	▮ Poor Note: If you select **Disable** for a specific peer, then the **Health** will show as ▮ Disabled. The peer health state is determined by the worse state of at least one statistic/parameter. For example, if at least one of the parameters is red, the health will be red. If there are only green and yellow indications and at least one of them is yellow, the health will be yellow. If all the parameters are green, the health parameter will be green. The peer health state is based on the following statistics (per last minute) which are displayed for a selected peer in the **Health Parameters** tab. There are three display options: ✓ – no threshold was reached, ! – a minor or major threshold was reached, ✖ – a critical threshold was reached. Note: As the ! icon relates to both minor and major thresholds, the color box surrounding the actual statistical number can be yellow or orange, depending if a minor or major threshold was reached. ▪ Error Answers – shows as ✓ or ✖ depending if the Error Answers Ratio is below/above the Error Thresholds. ▪ Error Answers Ratio (%.) – the number of error responses returned from the peer as compared to the number of requests sent to peer ▪ Error Thresholds – the allowed ratio between the number of requests sent to the peer and the number of error answers returned from the peer. Can be configured in **Topology>Peer Profiles>General** (see *Adding a Peer Profile*)**.** ▪ Timeouts - shows as ✓ or ✖ depending if the Timeouts Ratio is below/above the Timeout Thresholds. ▪ Timeouts Ratio (% per min.) – the number of requests not answered by the peer as compared to the number of requests sent to the peer

Column	Description
	▪ Timeout Thresholds – the allowed ratio between the number of requests sent to the peer and the number of requests not answered by the peer. Can be configured in **Topology>Peer Profiles>General** (see *Adding a Peer Profile*)**.** ▪ Requests Sent to Peer – shows as ✓ , ✗ , or ❗ depending if the requests sent are above/below the minor, major, critical threshold levels. The average number of requests sent per second from the SDC to a peer (over the last minute) ▪ Peer Outgoing Rate Limit – the user configured (**Topology**> **Peer Profiles>Rate Limit**, see *Configuring the Outgoing Traffic Rate Limits for Peers)* rate limit of sent messages from the SDC to system peers ▪ Requests Sent to Peer Thresholds – the defined minor, major, critical threshold levels ▪ Requests Received from Peer – shows as ✓ , ✗ , or ❗ depending if the requests received are above/below the minor, major, critical threshold levels. The average number of requests received per second by the SDC from a peer (over the last minute) ▪ Peer Incoming Rate Limit – the user configured (**Topology**>**Peer Profiles>Rate Limit**, see *Configuring the Incoming Traffic Rate Limits for Peers*) rate limit of received requests from the system peers to the SDC ▪ Requests Received from Peer Thresholds – the defined minor, major, critical threshold levels ▪ Round Trip Time – shows as ✓ or ✗ depending if the Round Trip Time is below/above the Round Trip Timeout Thresholds. ▪ Round Trip Time average processing time of requests by the peer (in milliseconds) measured from when the request is received by the peer from the SDC until the answer is sent from the peer to the SDC ▪ Round Trip Time Thresholds – the allowed roundtrip time frame in ms. ▪ Network Queue Usage – shows as ✓ , ✗ , or ❗ depending if the usage is above/below the minor, major, critical threshold levels. ▪ Network Queue Usage – the number of answers and requests waiting to be written to the machine socket

Column	Description
	▪ Network Queue Usage Thresholds – the defined minor, major, critical threshold levels ▪ Status: Corresponds to the value in the Status column. ▪ Active Alarms – link to view alarms currently raised for this peer in **Alarms>Active Alarms** Additional Information: ▪ Out Of Service Time– the percentage of time the peer was not available in the last minute ▪ Partial Out of Service Time– the percentage of time that the peer was operating in Message Prioritization mode in the last minute ▪ All peer Health statistics – links to the **Reports** screen to view statistics in different time resolutions
Status	Indicates whether the peer is currently connected to an SDC. ▪ When all FEP connections to the peer are open and all CPF processes are open or not available, the status is indicated as ✓ Open . ▪ When all FEP connections are open and at least one of the CPF processes is partially out of service, the status is indicated as ! Limited . In a CPF-only deployment, if one CPF process is open and other CPF processes are either closed, out of service, or pending connection, the status is indicated as limited. ▪ When all FEP connections to the peer are closed or when the FEP connections are open but all CPFs are out of service, the status is indicated as ✕ Closed Note: If you select **Disable** for a specific peer, then the **Status** will show as ● Disabled.
SDC Nodes	
Node Name	The name of the CPF or FEP that is connected to a selected peer

Column	Description
Health	Indicates the health (**Good**, **Fair**, **Poor, or Not Available**) of the peer per CPF. The health of each peer per CPF is determined by certain health statistics, as described above for peer health. The Node Health state can show as ❙ **Not Available**, when the peer has a **Not Available Status.**
Status	Indicates the status (**Open, Out of Service, Out of Service Partially**, and **Closed**) for the FEP and each CPF. For CPF-only deployments, there is also a **Pending** and **Not Available** status option.

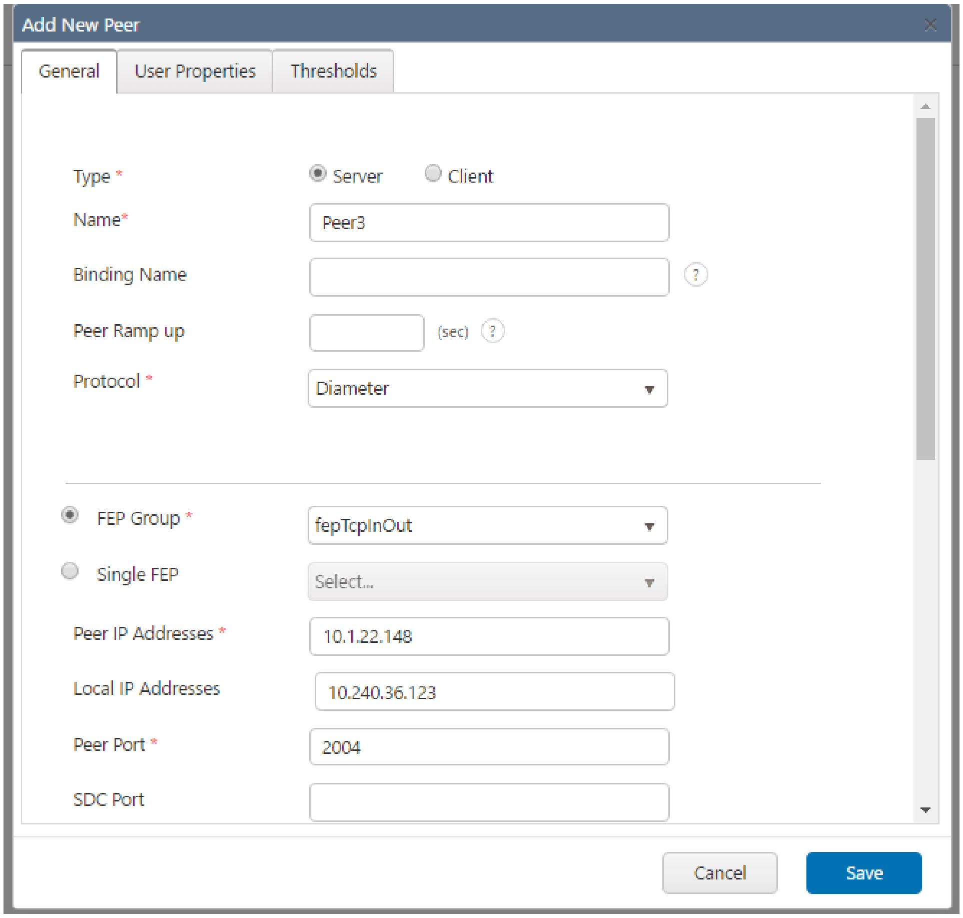
3.3.5.2 Adding a New Peer

This section describes how to add a new peer.

To add a new Peer:

1. Go to **Topology** > **Specific Site Settings** > <**Site Name**>> **Peers** and click **Add**.
The Add Peer window appears:

Figure 11: Add Peer Window



Type * ● Server ○ Client

Name* Peer3

Binding Name ?

Peer Ramp up (sec) ?

Protocol * Diameter ▼

● FEP Group * fepTcpInOut ▼

○ Single FEP Select... ▼

Peer IP Addresses * 10.1.22.148

Local IP Addresses 10.240.36.123

Peer Port * 2004

SDC Port

2.

a. In the **General** tab:

a. Under **Type**, define the peer as either a **Server** or **Client**.

In the **Name** field, enter a user-friendly display name to identify the peer. e.g. Server1.

Note: After submitting a new peer, its name may not be modified. The combined length of the peer name and pool name length cannot exceed 265 characters. As you cannot always view the peer name in advance, better not to use very lengthy peer and pool names. All (client) peer names must be unique, even to any previously deleted server peers.

In **Binding Name**, type in a name used by the routing mechanism to bind sessions belonging to this peer with other sessions.

b. In **Protocols**, out of the available signaling protocols, select the protocol used by the peer from the drop-down list.

c. The wizard configuration options display according to your protocol selection. Continue to the next section according to your protocol selection.

Note: The SS7 protocol is only supported in bare-metal deployments.

3. Under the **User Properties** tab:

You can create additional properties for the Peer Profile and define the value for these properties. These properties can be used in the Peer Profile scripts and decision table.

a. Click **Add**.

i. In the **Name** field, enter a user friendly property name.

ii. In the **Value** field, enter the desired value for the property.

iii. In the **Path** field, the path name for the property is displayed.

Note: The path name is only displayed once the peer is added.

User properties can also be defined using the setEntityProperties Web Service API method and retrieved using the getEntityProperties Web Service API method or using Groovy scripting. For more information about the Web Service API methods, see the *F5 SDC Web Services API Guide*.

4. Under **Thresholds**, you can configure the threshold percentages for generating rate limit alarms by severity level (**Critical**, **Major**, **Minor)** for the **Outgoing and Incoming TPS vs Rate Limit** per peer. For more information about how threshold management is part of overload control, see *Configuring Alarm Thresholds* and for how it can be configured globally, see 🌐 *Threshold Management*.

3.3.5.2.1 Diameter Peer

This section continues with the wizard configuration for adding a Diameter peer.

Note: You are required to define the FEP Group, Peer IP Addresses and the Peer Port parameters.

To add a server peer with Diameter properties:

1. In **FEP Group**, select a FEP group (fep-in/fep-out), from the drop-down list to which the peer is connecting.

   a. Alternatively, in **Single FEP**, select a FEP node from the drop-down list to which the peer is connecting.

2. In **Peer IP Addresses**, set the address (single or multiple) where the peer (client or server) is hosted.

3. In Local IP addresses, set the address (single or multiple) for the FEP.

   Note: This is not a mandatory field, but is recommended for SCTP message transport.

4. In the **Peer Port** field, specify an available port number for the peer to access the SDC Components.

5. In **SDC Port**, set the local port from which to send messages to a Server Peer.

6. In **Peer Profile**, you may choose to assign a special attribute to the remote peer. Peers assigned with a Peer Profile are handled in a predefined manner – they may receive unique messages and send unique messages.

   Note: For additional information on Peer Profiles, see *Configuring Peer Profiles*.

7. In **Idle Connection Time (TW) (Millis)**, set the time (in milliseconds) that the connection will remain open, without any traffic, before sending a watchdog request.

Note: The default value is 30,000 ms. The server will not be reconnected until traffic resumes. This parameter is only relevant in keep alive mode.

8. In **Reestablish Connection Time (TC)**, set the time (in milliseconds) for reconnecting the peer, after the connection is disconnected by the server.

Note: This parameter is only relevant when in keep alive mode and the value must be between 1-30000 milliseconds.

9. In **Local Host**, set a value for the message’s origin-host.

10. In **Local Realm**, set a value for the message’s origin-realm.

11. In **Weight**, set the peer’s weight (a number) in traffic distribution, in case it is included in a Weighted Round Robin Pool or in Contextual Pool load balancing policy.

Note: For additional information on Weighted Round Robin and Contextual and other load balancing policies, see *Assigning a Load Balancing Policy*.

12. In **Priority**, set the Server’s position in a pool’s activation and server selection procedures.

Note: When selecting a Queue Size Ratio load balancing policy (see *Assigning a Load Balancing Policy)*, do not set the **Priority** parameter.

For more information on pool’s activation and server selection procedure, see *Configuring Pools*.

13. Select **Use SCTP Transport** to use SCTP (rather than TCP) for message transport.

Select **Use for Geo Redundant Sites Connection** to enable the peer to handle proxied requests.

---

14. Note: When enabling **Use for Geo Redundant Sites Connection**, Geo-Site Loop Detection is automatically enabled. The SDC adds a Proxy-From-Replicator AVP to a Diameter message that is proxied between two sites. In the event the geo- redundant site cannot send the proxied message to the peer, the message is discarded so that SDC performance is not impacted.

---

15. Select **Use single Diameter identity in dual active-active FEP mode** to enable the peer to use a single identity on multiple FEPs according to the configured Local Host.

16. Click **Save**. The new peer is displayed in the Peer table.

3.3.5.2.2 RADIUS Peer

This section continues with the wizard configuration for adding a RADIUS peer.

Note: You are required to define the FEP Group, Peer IP Addresses and the Peer Port parameters.

---

1. **To add a server peer with RADIUS properties:**

2. In **FEP Group**, select a FEP group (fep-in/fep-out), from the drop-down list to which the peer is connecting.

3. Alternatively, in **Single FEP**, select a FEP node from the drop-down list to which the peer is connecting.

4. In **Peer IP Addresses**, set the address (single or multiple) where the peer (client or server) is hosted.

In Local IP addresses, set the address (single or multiple) for the FEP.

Note: This is not a mandatory field, but is recommended for SCTP message transport.

In **Peer Port**, specify an available port number for the peer to access the SDC Components.

5. In **SDC Port**, set the local port from which to send messages to a Server Peer.

6. 7. In **Peer Profile**, you may choose to assign a special attribute to the remote peer. Remote Peers assigned with a Peer Profile are handled in a predefined manner – they may receive unique messages and send unique messages.

Note: For additional information on Peer Profiles, see *Configuring Peer Profiles*.

8. In **Reestablish Connection Time (TC)**, set the time (in milliseconds) for reconnecting the peer, after the connection is disconnected by the server

Note: This parameter is only relevant when in keep alive mode and the value must be between 1-30000 milliseconds.

9. In **Weight**, set the peer's weight (a number) in traffic distribution, in case it is included in a Weighted Round Robin Pool or in Contextual Pool load balancing policy.

10. Note: For additional information on Weighted Round Robin and Contextual and other load balancing policies, see *Assigning a Load Balancing Policy*.

In **Priority**, set the Server's position in a pool's activation and server selection procedures.

Note: When selecting a Queue Size Ratio load balancing policy (see *Assigning a Load Balancing Policy)*, do not set the **Priority** parameter.

For more information on pool’s activation and server selection procedure, see *Configuring Pools*.

11. In **Shared Secret**, type in the text string that serves as a password between the Remote RADIUS Client and the RADIUS Virtual Server. The shared secret is used to verify that both client and server are using the same “password”. It is also used to verify that the RADIUS message has not been modified when sent and to encrypt RADIUS attributes.

In **Connection Pool Size**, set the maximum number of open RADIUS connections.

12. Click **Save**. The new peer is displayed in the Peers.

13.

3.3.5.2.3 HTTP Peer

This section continues with the wizard configuration for adding an HTTP peer

Note: You are required to define the Peer IP Addresses and the Peer Port parameters.

To add a server peer with HTTP properties:

1. In **Peer IP Addresses**, set the address (single or multiple) where the peer (client or server) is hosted.

2. In Local IP addresses, set the address (single or multiple) for the FEP.

3. Note: This is not a mandatory field, but is recommended for SCTP message transport.

4. In **Peer Port**, specify an available port number for the peer to access the SDC Components.

In **Peer Profile**, you may choose to assign a special attribute to the peer. Remote Peers assigned with a Peer Profile are handled in a predefined manner – they may receive unique messages and send unique messages.

Note: For additional information on Peer Profiles, see *Configuring Peer Profiles*.

In **Max Connection Count Limit (Per Server)**, set the maximum number of open HTTP connections.

5. In **Idle Connection Time (TW) (Millis)**, set the time (in milliseconds) that the connection will remain open, without any traffic.

6. Note: The default value is 30,000 milliseconds. The server will not be reconnected until traffic resumes. This parameter is only relevant when in keep alive mode.

7. In Reestablish Connection Time (TC), set the time (in milliseconds) for reconnecting the peer, after the connection is disconnected by the server.

Note: This parameter is only relevant when in keep alive mode and the value must be between 1-30000 milliseconds..

8. In **Weight**, set the peer's weight (a number) in traffic distribution, in case it is included in a Weighted Round Robin Pool or in Contextual Pool load balancing policy.

9. Note: For additional information on Weighted Round Robin and Contextual and other load balancing policies, see *Assigning a Load Balancing Policy*.

In **Priority**, set the Server's position in a pool's activation and server selection procedures.

Note: When selecting a Queue Size Ratio load balancing policy (see *Assigning a Load Balancing Policy)*, do not set the **Priority** parameter.

For more information on pool's activation and server selection procedure, see *Configuring Pools*.

Select **Keep Alive** to preserve a persistent HTTP connection.

Select **Use for Geo Redundant Sites Connection** to enable the peer to handle proxied requests.

10.

11. Note: Only one peer per site can be configured as a proxy peer, to be used for proxying requests between sites (by selecting the **Use for Geo Redundant Sites Connection** checkbox).

Note: When enabling **Use for Geo Redundant Sites Connection**, Geo-Site Loop Detection is automatically enabled. The SDC adds a Proxy-From-Replicator AVP to a HTTP message that is proxied between two sites. In the event the geo- redundant site cannot send the proxied message to the peer, the message is discarded so that SDC performance is not impacted.

12. Click **Save**. The new peer is displayed in the Peers table.

3.3.5.2.4 LDAP Peer

This section continues with the wizard configuration for adding an LDAP peer.

Note: You are required to define the Peer IP Addresses and the Peer Port parameters.

1. **To add a peer with LDAP properties:**

2. In **Peer IP Addresses**, set the address (single or multiple) where the peer (client or server) is hosted.

In Local IP addresses, set the address (single or multiple) for the FEP.

3. Note: This is not a mandatory field, but is recommended for SCTP message transport.

In Peer **Port**, specify an available port number for the peer to access the SDC Components.

In **Peer Profile**, you may choose to assign a special attribute to the peer. Peers assigned with a Peer Profile are handled in a predefined manner – they may receive unique messages and send unique messages.

4.

Note: For additional information on Peer Profiles, see *Configuring Peer Profiles*.

In **Bind User**, type in the LDAP user for the directory server authentication.

5. In **Bind Password**, type in the LDAP user's password for the directory server authentication.

6. In **Reestablish Connection Time (TC)**, set the time (in milliseconds) for

7. reconnecting the peer, after the connection is disconnected by the server.

Note: This parameter is only relevant when in keep alive mode and the value must be between 1-30000 milliseconds.

8. In **LDAP Pool Size**, specify the number of connections to use while connecting the LDAP remote peer.

9. In **Weight**, set the peer's weight (a number) in traffic distribution, in case it is included in a Weighted Round Robin Pool or in Contextual Pool load balancing policy.

10. Note: For additional information on Weighted Round Robin and Contextual and other load balancing policies, see *Assigning a Load Balancing Policy*.

In **Priority**, set the Server's position in a pool's activation and server selection procedures.

Note: When selecting a Queue Size Ratio load balancing policy (see *Assigning a Load Balancing Policy)*, do not set the **Priority** parameter.

For more information on pool’s activation and server selection procedure, see *Configuring Pools*.

Click **Save**. The new peer is displayed in the Peers table.

3.3.5.2.5 File Peer

11. This section continues with the wizard configuration for adding a File peer.

Note: This protocol is not supported in this release.

You are required to define the Primary IP and Primary Port parameters.

To add a File server peer:

1. Set the **Primary IP** of the File Server.
2. Set the **Primary Port** of the File Server.
3. Set the **Secondary IP** of the File Server.
4. Set the **Secondary Port** of the File Server.
5. In **Split By**, set the value for which the messages will be divided into groups.
6. In **Number of Groups**, set how many groups will be needed.
7. In **Reestablish Connection Time (TC)**, set the time (in milliseconds) for reconnecting the peer, after the connection is disconnected by the server.

8. Note: This parameter is only relevant when in keep alive mode and the value must be between 1-30000 milliseconds..

In **Weight**, set the peer’s weight (a number) in traffic distribution, in case it is included in a Weighted Round Robin Pool or in Contextual Pool load balancing policy.

Note: For additional information on Weighted Round Robin and Contextual and other load balancing policies, see *Assigning a Load Balancing Policy*.

9. In **Priority**, set the Server's position in a pool's activation and server selection procedures.

Note: When selecting a Queue Size Ratio load balancing policy (see *Assigning a Load Balancing Policy)*, do not set the **Priority** parameter.

For more information on pool's activation and server selection procedure, see *Configuring Pools*.

10. Click **Save**. The new peer is displayed in the Peers table.

3.3.5.2.6 SS7 Peer

This section continues with the wizard configuration for adding an SS7 peer.

Note: Only one SS7 peer can be added per CPF. If you try to add more than one SS7 peer, the following message appears "Cannot add more than one SS7 Peer. 'server_SS7 already exists."

Note: SS7 peers can only be defined as server peers.

Note: SS7 is only supported in bare metal deployments.

To add a server peer with SS7 properties:

1. In **Peer Profile**, you may choose to assign a special attribute to the remote peer. Remote peers assigned with a Peer Profile are handled in a predefined manner – they may receive unique messages and send unique messages.

2. In **GSM MAP handling** select Automatic or Manual to define how messages will be transformed.

When GSM MAP handling, the **Application Protocol** option is enabled. In **Application Protocol**, type the protocol of the expected messages.

In **Map Version**, type in the SS7 version of the expected messages.

3. Select **Route on Global Title** to route messages of this peer using its global title

4. indicator (the SS7 IP equivalent).

5. In **Encoding Scheme**, type one for odd numbers or two for even numbers.

a. In **Global Title Indicator**, type in 4 for Global Title format.

b. In **Nature of Address Indicator**, type three for national addresses or four

c. for international addresses.

In **Translation Type**, type in 0.

d. **Global Title Address**, type in the peer’s address (maximum length is 15

e. digits)

f. In **Numbering Plan**, type in 1 for ISDN/telephony numbering plan (E.164) or 7 for ISDN/Mobile Numbering Plan (E.214).

6. Click **Save**. The new peer is displayed in the Peers table.

3.3.5.2.7 Editing a Peer

This section describes how to edit a peer.

1. Note: You may only edit Server Peers.

2. **To edit a peer:**

Select a peer from the Peer table list and click **Edit**. The Edit Peer wizard appears.

You may edit the enabled fields, as detailed in *Adding a New Peer*.

Note: **Name**, **Protocol** and **Type** parameters cannot be edited.

3.3.5.3 Removing a Peer

This section describes how to remove any of the peers from the Peers table.

To remove a peer from the Peers table:

1. Select the row of the peer you wish to remove.
2. Click **Remove**.

   A confirmation message appears.
3. Click **OK**.

Note: When a peer is a part of a pool and it is removed, it is also removed from the pool. For more information on pools, see *Configuring Pools*.

3.3.6 Configuring Pools

Pools are groups containing peers. Using the pool configuration, the SDC decides how to assign policies, such as, load balancing, or overload control. A load balancing policy is assigned when you want messages sent to a pool to be routed to peers according to specific rules.

This section describes how to do the following:

- *Viewing a List of Pools*
- *Adding a New Pool*
- *Assigning a Load Balancing Policy*
- *Assigning a Load Balancing Policy between Pools*
- *Assigning a Broadcast Pool Policy*
- *Configuring a Notification Pool*
- *Configuring Pool Failover Policy*
- *Configuring Message Prioritization*

- *Editing a Pool*
- *Removing a Pool*

3.3.6.1 Viewing a List of Pools

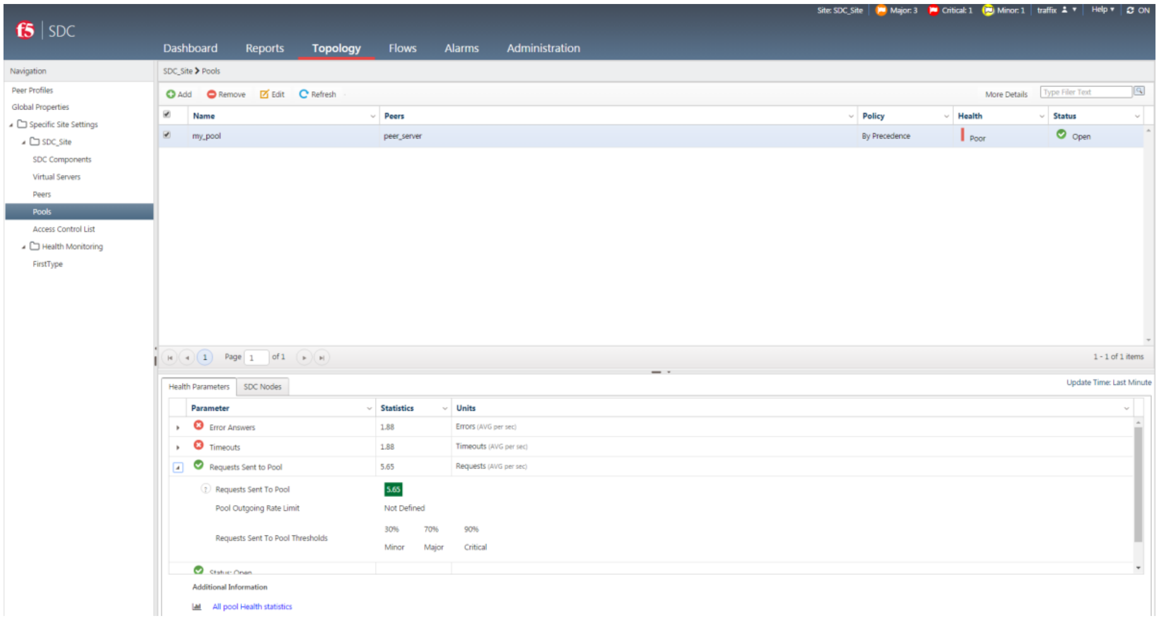
You can view the current list of configured pools.

To view the list of pools:

Go to **Topology** > **Specific Site Settings** > **<Site Name>** > **Pools**.

1.

Figure 12: Pools



Name	Peers	Policy	Health	Status
my_pool	peer_server	By Precedence	Poor	Open

Parameter	Statistics	Units
Error Answers	1.88	Errors (AVG per sec)
Timeouts	1.88	Timeouts (AVG per sec)
Requests Sent to Pool	5.65	Requests (AVG per sec)
Requests Sent To Pool	5.65	
Pool Outgoing Rate Limit	Not Defined	
Requests Sent To Pool Thresholds	30% Minor 70% Major 90% Critical	
Status: Open		

Table 20 presents a list of a pool's properties.

Table 19: Pool's Properties

Column	Description
Name	A user-friendly display name assigned to the pool. e.g. Pool1
Peers	The list of peers which are included in the pool. e.g.: server1, server2.
Policy	The method by which messages are routed within the pool. For example, load balancing policy of Weighted Round Robin.

Column	Description
	Note: For more information on these policies, see *Assigning a Load Balancing Policy.*
Health	Indicates the health status of each pool. The health of the pool is based on the health of the peers, as defined by certain health statistics, within the pool. Note: You can click **More details** to view the health related statistics for all the pools. Peers that are manually disabled by the user in the Web UI are not included in the pool health calculation. The pool health is presented with one of three possible states: 🟩 Good 🟨 Fair 🟥 Poor The pool health state is determined by the worse state of at least one parameter. For example, if at least one of the parameters is red, the health will be red. If there are only green and yellow indications and at least one of them is yellow, the health will be yellow. If all the parameters are green, the health parameter will be green. The pool health state is based on the following statistics (per the last minute) which can be viewed in the bottom pane **Health Parameters** tab by selecting one of the pools. There are three display options: ✅ – no threshold was reached, ❗ – a minor or major threshold was reached, ❌ – a critical threshold was reached. Note: As the ❗ icon relates to both minor and major thresholds, the color box surrounding the actual statistical number can be yellow or orange, depending if a minor or major threshold was reached. ▪ Error Answers – shows as ✅ or ❌ depending if the Error Answers Ratio is below/above the Error Thresholds. ▪ Error Answers Ratio (%.) – the number of error responses returned from the pool as compared to the number of requests sent to the pool ▪ Error Thresholds – The allowed ratio between the number of error answers returned from the pool number as compared to the number of requests sent to the pool.

Column	Description
	▪ Timeouts – shows as ✓ or ✗ depending if the Timeouts Ratio is below/above the Timeout Thresholds. ▪ Timeouts Ratio (% per min.) – the number of requests not answered by the pool as compared to the number of requests sent to the pool ▪ Timeout Thresholds – the allowed ratio between the number of requests not answered by the pool as compared to the number of requests sent to the pool. Note: When the Timeout Threshold passes the configured percentage, the peerHealthTimeouts alarm is triggered, which then impacts upon the peer health and the relevant pool health. This same alarm is also triggered when the Busy Error Answers Threshold is passed. The peerHealthTimeouts alarm message description (**Alarms** > **Active Alarms**) describes both thresholds as a "timeout," and consequently, can be a reference to the Timeout Threshold or the Busy Error Answers Threshold. These thresholds can be configured from the peer profile, peer or pool (Topology) configurations. ▪ Requests Sent to Pool – shows as ✓ , ✗ , or ⚠ depending if the requests sent are above/below the minor, major, critical threshold levels. The average number of requests sent per second from the SDC to a pool (over the last minute) ▪ Pool Outgoing Rate Limit – the user configured (**Topology**> **Pools>General**, see *Configuring the Outgoing Traffic Rate Limits for Pools*) rate limit of sent messages from the SDC to pools ▪ Requests Sent to Pool Thresholds – the defined (**Topology**> **Pools>Thresholds**) minor, major, critical threshold levels ▪ Status: Corresponds to the value in the Status column. ▪ Active Alarms – link to view alarms currently raised for this peer in **Alarms>Active Alarms** Additional Information: ▪ All peer Health statistics – links to the **Reports** screen to view statistics in different time resolutions
Status	Indicates the availability of the pool.

Column	Description
	▪ When all CPFs consider the pool as open. "Open" means that at least x defined number of peers of the pool are open at the CPF, the status is indicated as ✓ Open. ▪ When a pool is open for some CPFs and out of service for other CPFs, the status is indicated as ! Limited. ▪ When all the CPFs are out of service for the pool, the status is indicated as ✖ Closed.
Node Health	Indicates the health (**Good**, **Fair**, or **Poor**) of the pool per connected CPF. The health of the pool is based on the health of the peers, as defined by certain health statistics, within the pool. In addition, the Health state can show as ▯ **Not Available**, when there is no available information.
Node Status	Indicates the status (**Open**, **Out of service**, **Partially Out of Service**, , **Closed, or Not Available**) of CPF.

3.3.6.2 Adding a New Pool

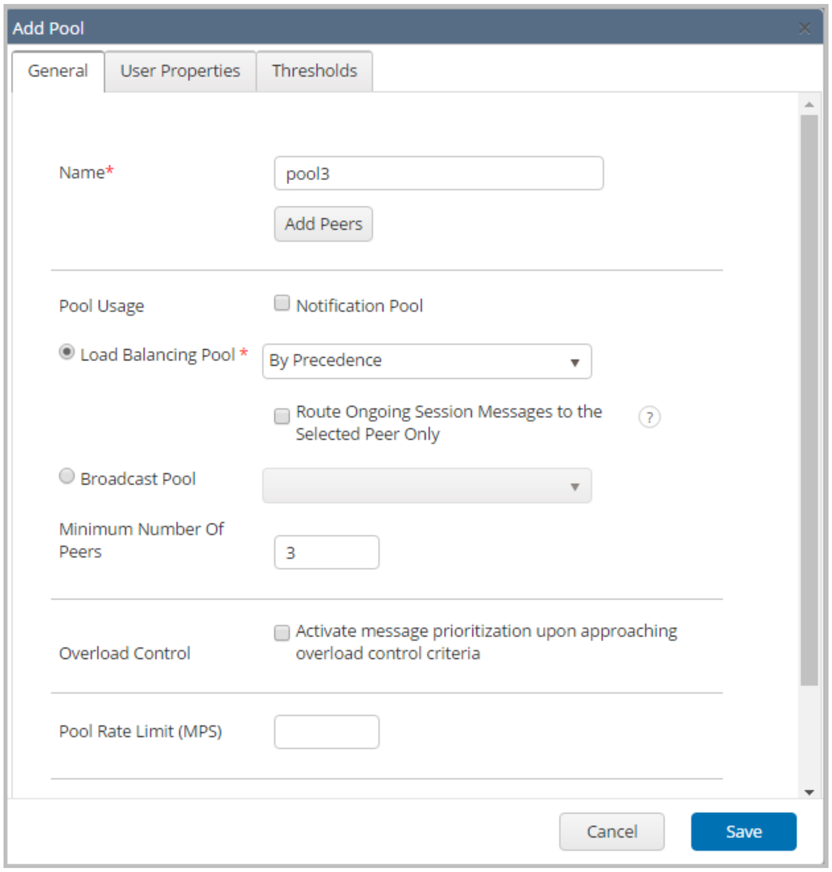
You can add a new pool and define which server peers belong to the added pool.

To add a new pool:

1. Go to **Topology** > **Pools** > **Add**.

The Add Pool dialog box is displayed:

Figure 13: Add Pools



Name* pool3

Pool Usage ☐ Notification Pool

● Load Balancing Pool * By Precedence

☐ Route Ongoing Session Messages to the Selected Peer Only ?

○ Broadcast Pool

Minimum Number Of Peers 3

Overload Control ☐ Activate message prioritization upon approaching overload control criteria

Pool Rate Limit (MPS)

2.

a. In the **General** tab:

In the **Name** field, enter a user friendly display name to identify the pool.

Note: After submitting a new Pool, its name may not be modified.

b. Pool names are case sensitive. The combined length of the peer name and pool name cannot exceed 265 characters. As you cannot always view the peer name in advance, better not to use very lengthy peer and pool names.

Click **Add Peers** to select the peer(s) that you want to include in the pool.

i. In the Peer Selection window, from the **Select peers…** drop-down, select the static peers that you want to include the pool. Under **Dynamically**

add Peers matching Peer Profiles, select the dynamic peers that you want to include the pool.

Note: To be able to dynamically add peers assumes that you have configured the relevant dynamic peer profiles in **Topology** > **Peer Profiles** >**Dynamic Peer Profiles**.

ii. Click **Save**.

c. In **Minimum Number of Peers**, enter the number of peers that are available.

The **Minimum Number of Peers** value determines the minimum number of servers that must remain available for traffic to be directed to a pool. If the number of open servers drops under this number, the pool will not be available for traffic and events will be routed to next available pool on the routing row.

Note: When no peer in the pool is available, an Error event occurs. The default value of **Minimum Number of Peers** is 1.

d. Under Pool Usage, you can configure the pool as follows:

i. In **Notification Pool** if you want messages to be copied to the pool. For more information, see *Configuring a Notification Pool*.

ii. In **Load Balancing Pool**, select the policy you wish to assign to the server peers included in the pool. For more information on the different Load Balancing polices, see *Assigning a Load Balancing Policy*.

iii. In **Broadcast Pool**, so that messages sent to a pool are routed to all of the server peers in the pool. For more information, see *Assigning a Broadcast Pool Policy*.

iv. Select **Route Ongoing Session Messages to the Selected Peer Only**, if you want to configure the pool for peer failover policy. For more information, *see Configuring Peer Failover Policy*.

e. Select **Activate message prioritization upon approaching overload control criteria** to have the pool prioritize processing of incoming messages among its peers when it is in an overloaded state. For more information, see *Configuring Overload Control Policy*.

f. In Pool **Rate Limit (MPS)**, enter the maximum messages per second that can be processed by the pool.

g. In **Ramp Up Split By**, enter the message property that the messages will be divided according to.

h. In **Pool Ramp-Up Time (Seconds)**, enter the time (in seconds), that the pool will be in ramp-up mode from when the mode is activated.

Note: Configuring Pool Ramp-Up Time helps prevent pool overload by limiting the message traffic to the pool during initialization.

After configuring the ramp-up mode in the Web UI, it can only be activated through the Web Service API by running setEntityProperties method with the following input parameter values:

Pathname – the path to the pool that is selected to be in ramp-up mode (i.e. Site/Site-name/Pool/Pool-name)

Key – “RampUp”

Value – “1” to activate

i.

For more information on how to configure these input parameters, see setEntityProperties in the *F5 SDC Web Services Guide*.

Click **Save**.

In the **User Properties** tab:

3. Using the User Properties, you can create additional properties for the pool and define the value for these properties. These properties can be used in scripts relating to the pool. Once defined, using these properties in scripts will reflect the specific value you defined.

i. In the **Name** field, enter a user friendly property name.

ii. In the **Value** field, enter the desired value for the property.

iii. In the **Path** field, the property's path name is displayed.

iv. Click **Save**.

Note: User properties can also be defined using the setEntityProperties Web Service API method and retrieved using the getEntityProperties Web Service API method or using Groovy scripting. For more information about the Web Service API methods, see the *F5 SDC Web Services API Guide*.

4. Under **Thresholds**, you can configure the threshold percentages for generating rate limit alarms by severity level (**Critical**, **Major**, **Minor)** for the **Outgoing TPS vs Rate Limit** per pool. For more information about threshold management For more information about how threshold management is part of overload control, see *Configuring Alarm Thresholds* and for and how it can be configured globally, see *Threshold Management*.

3.3.6.3 Assigning a Load Balancing Policy

Load Balancing policies are used when messages sent to a pool are routed to one of the pool's peers. The peer selection is based on the pool's defined load balancing policy. The following sections detail the different policies according to which SDC's load balancing mechanism may operate, explains the differences between them, and describes the state in which each policy should be used.

To assign a Load Balancing Pool policy:

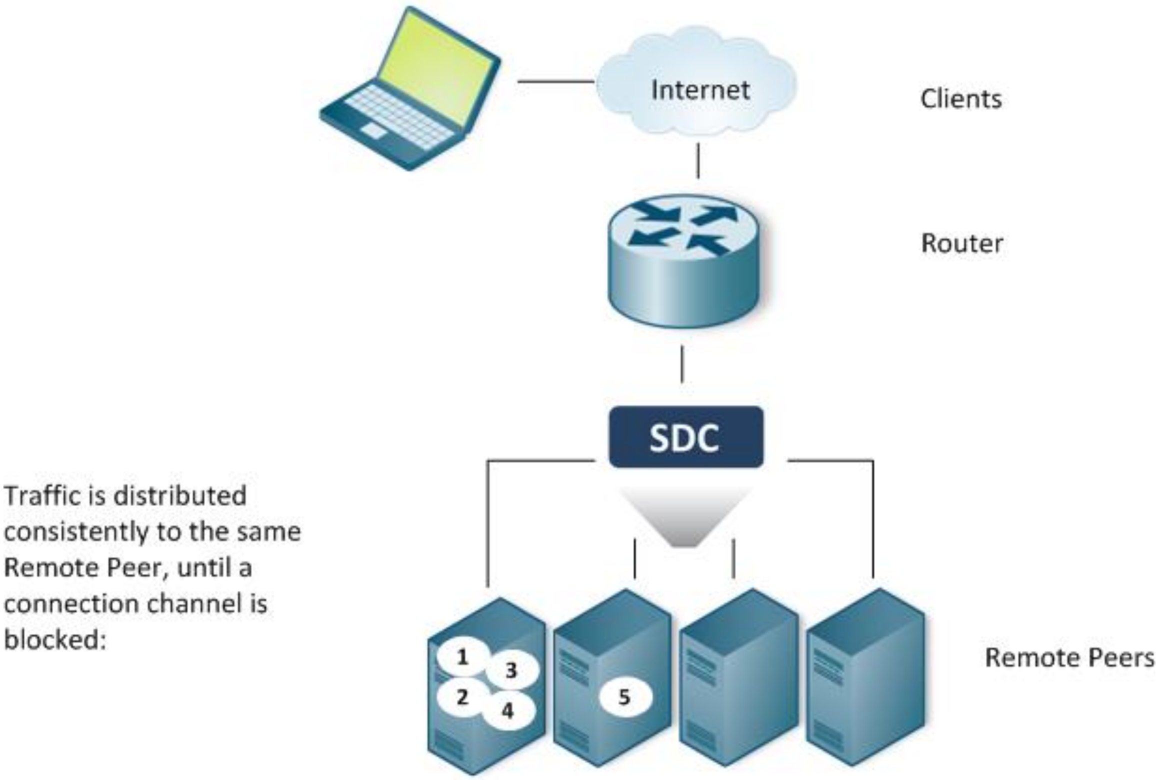
Go to **Topology** > **Pools** > **Add/Edit** > **General** > **Load Balancing Pool**.

Select the relevant policy from the drop-down list.

1. ### 3.3.6.3.1 By Precedence
2. When selecting the **By Precedence** policy, messages are sent to the first server peer in the pool until a connection channel is blocked. When the connection channel to the first server peer in the pool is blocked, the message is sent to the next server peer in the pool, etc. When the connection channel is unblocked, the messages are redirected to the first server peer.

Incoming requests are distributed as shown in *Figure 15*.

Figure 14: By Precedence Policy


Internet
Clients
Router
SDC
Traffic is distributed consistently to the same Remote Peer, until a connection channel is blocked:
1
3
2
4
5
Remote Peers

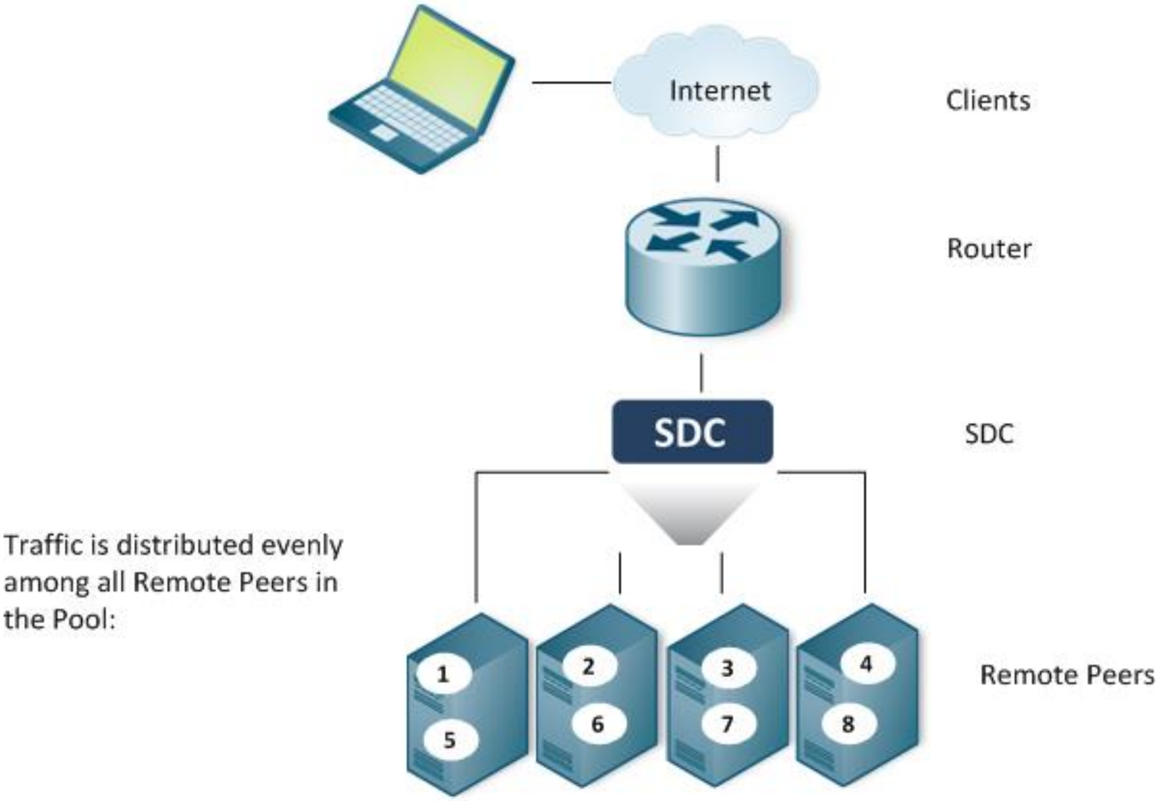
3.3.6.3.2 Round Robin

When selecting the **Round Robin** policy, traffic is evenly distributed across the pool's available server peers and the server peer to which the new request is delivered is the next available in line.

Round Robin is a static algorithm, no external parameters are taken under account upon request distribution.

Incoming requests are distributed as shown in *Figure 16*.

Figure 15: Round Robin Policy

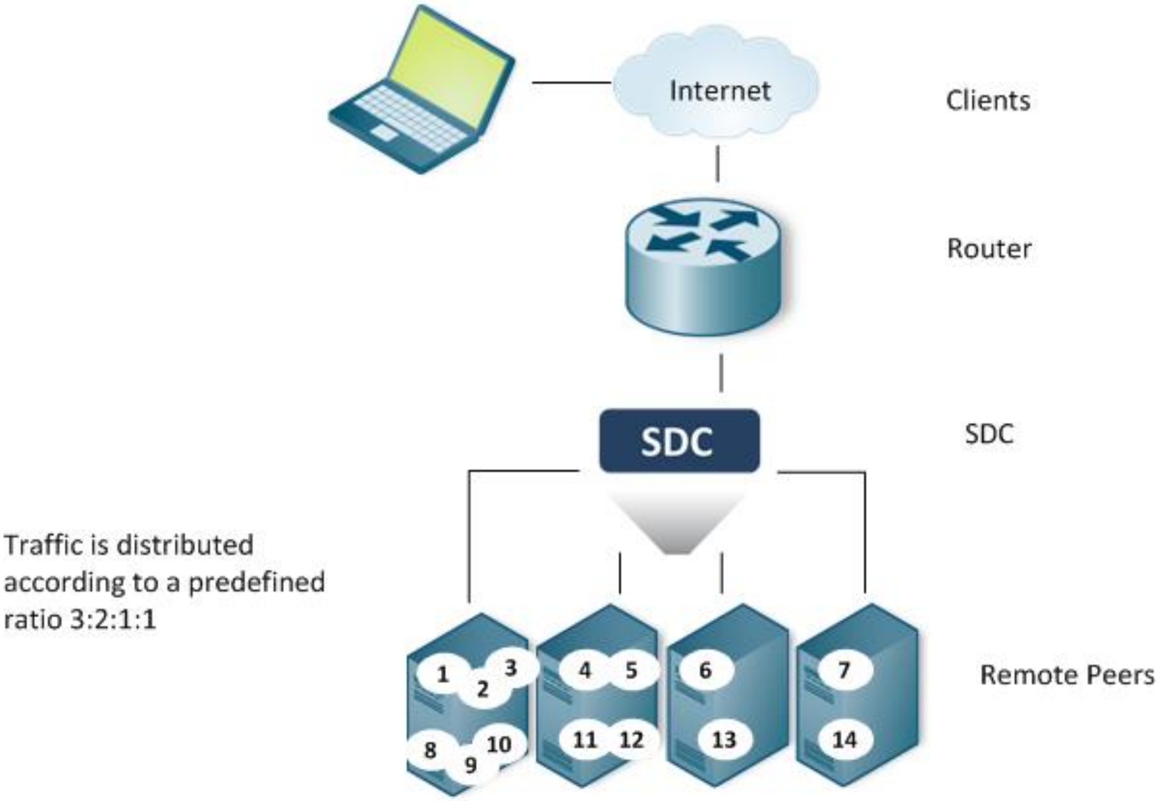

Internet
Clients
Router
SDC
SDC
Traffic is distributed evenly among all Remote Peers in the Pool:
1
5
2
6
3
7
4
8
Remote Peers

3.3.6.3.3 Weighted Round Robin

When selecting the **Weighted Round Robin** policy, traffic is distributed across the pool's available server peers according to a predefined proportion. The weight of each server peer is set when establishing it and should be based upon its ability to handle incoming requests. Weighted Round Robin is a static algorithm. No external parameters are taken under account upon request distribution.

With Weighted Round Robin, new requests are distributed in the Round Robin pattern, but instead of sending the request to the next available Server Peer in line, requests are sent to the Server Peer that had not yet reached its quota. When repeating requests of an already known session (e.g.:Accounting-Record-Type STOP after Accounting-Record-Type START), the policy's calculation is not performed and the second request is sent to the same server as the previous one. When one of the Server Peers fails to handle the request, the second request will be sent based on the session's history. When the set ratio is 3:2:1:1 incoming requests are distributed as shown in *Figure 17*.

Figure 16: Weighted Round Robin Policy


Internet
Clients
Router
SDC
SDC
Traffic is distributed according to a predefined ratio 3:2:1:1
1
2
3
8
9
10
4
5
11
12
6
13
7
14
Remote Peers

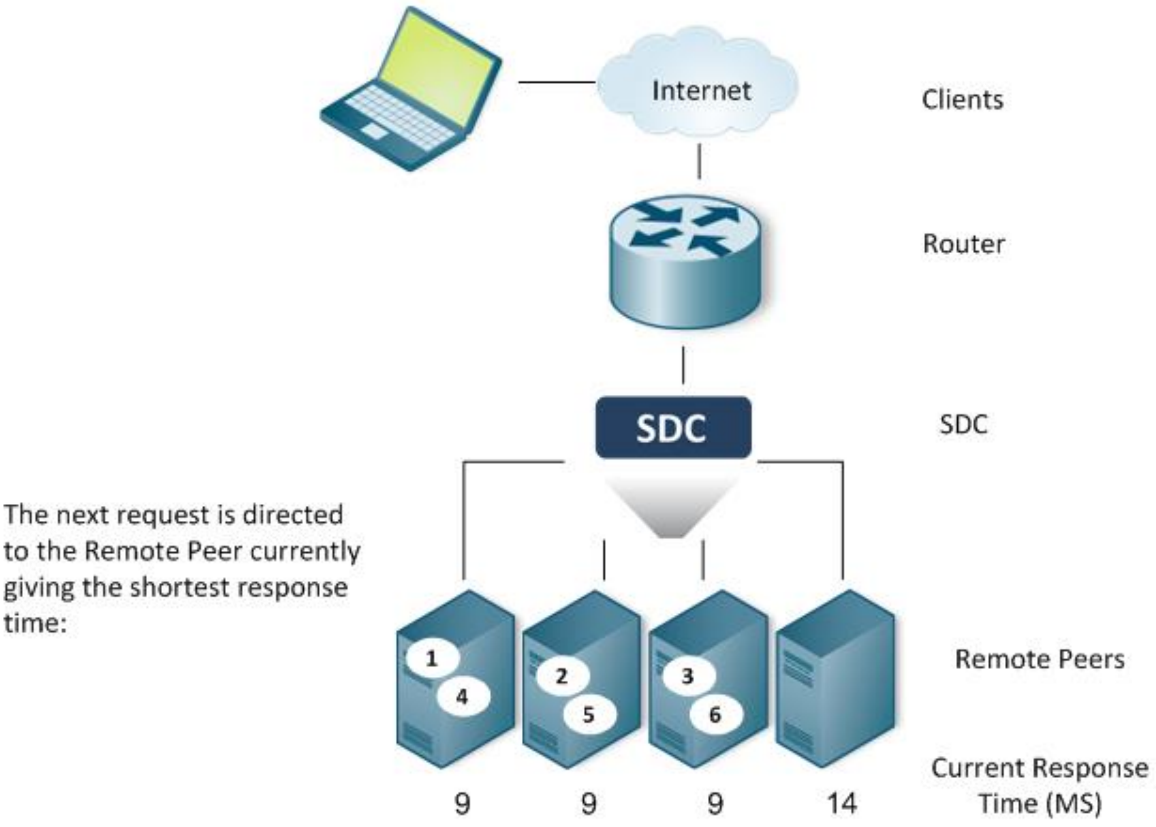
3.3.6.3.4 Fastest Response Time

When selecting a **Fastest Response Time** policy, requests are sent to the server peers according to their response time. The response time is used as the weight of the Remote Server. Remote Server static configured weight is ignored.

Fastest Response Time is a dynamic algorithm since it takes external parameters (response time) under account upon request distribution.

Incoming requests are distributed as shown in *Figure 18*.

Figure 17: Fastest Response Time Policy


Internet
Clients
Router
SDC
SDC
The next request is directed to the Remote Peer currently giving the shortest response time:
1
4
2
5
3
6
Remote Peers
9
9
9
14
Current Response Time (MS)

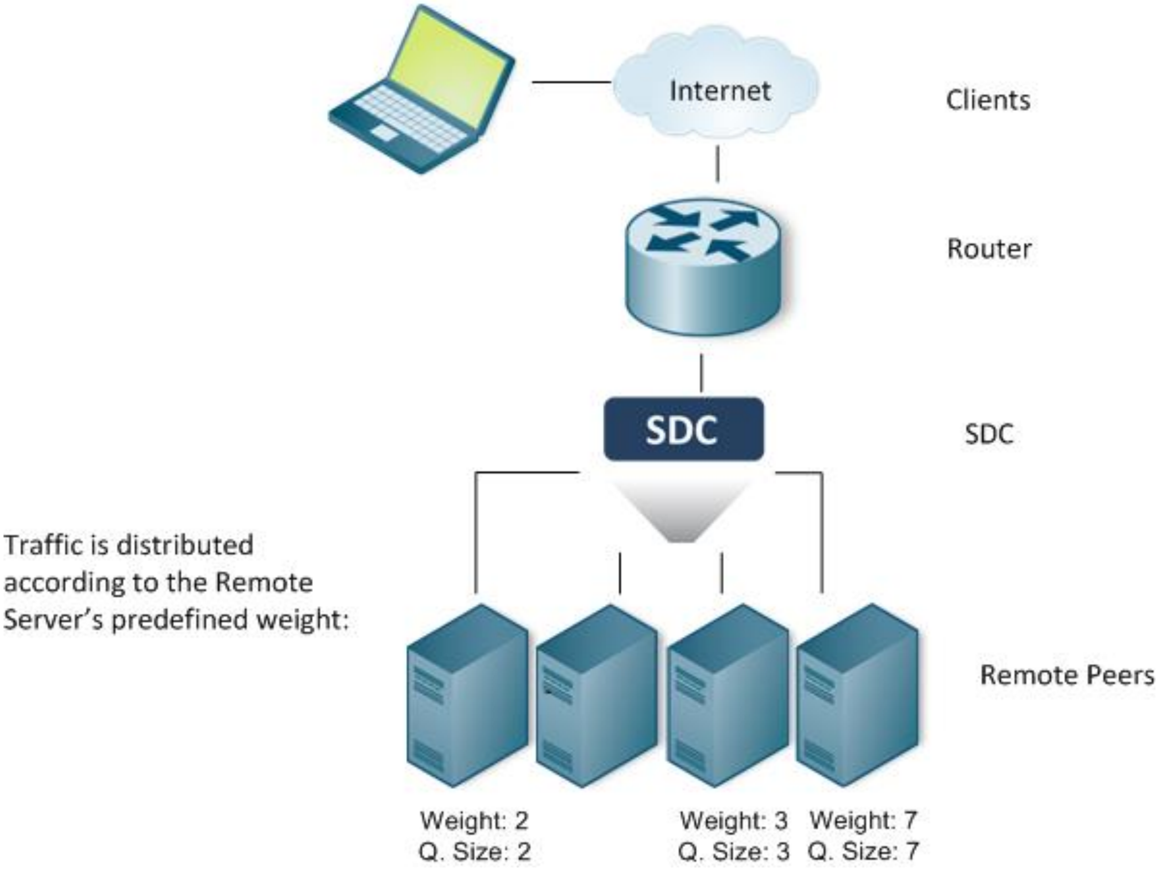
3.3.6.3.5 Queue Size Ratio

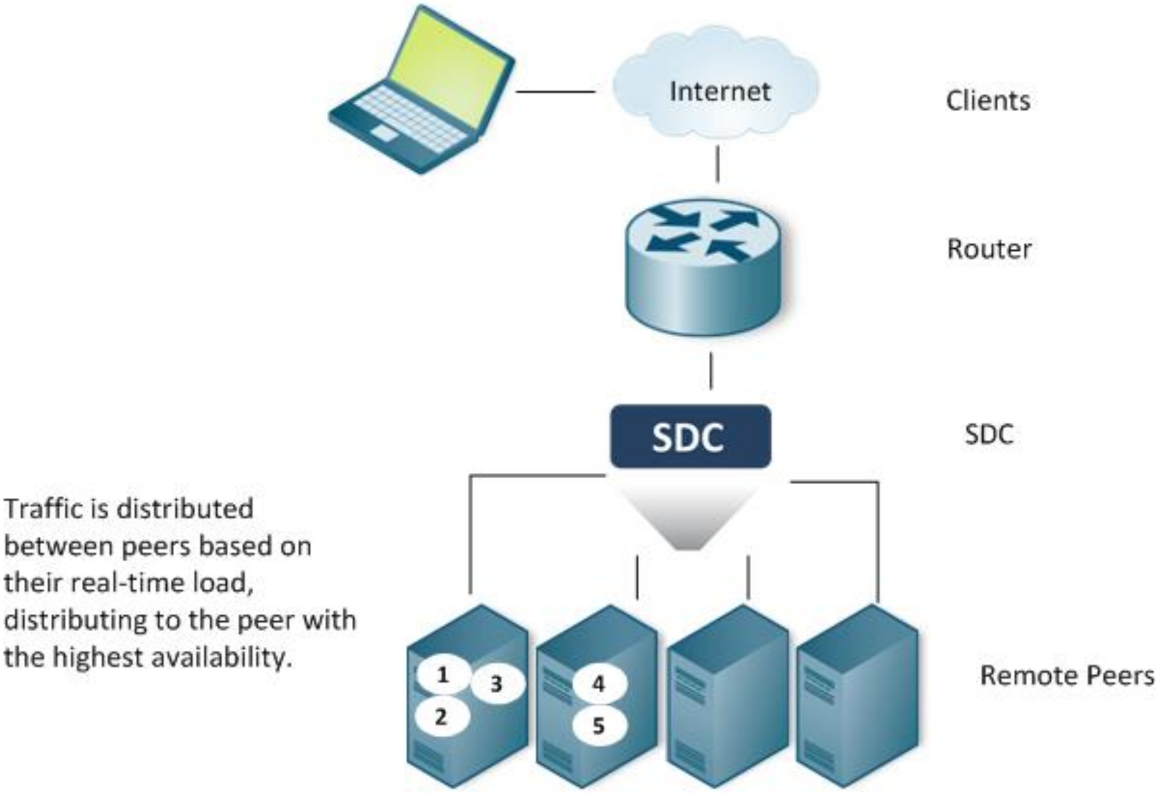
When selecting a **Queue Size Ratio** policy, the SDC distributes the requests to the Remote Servers according to the weight/queue length ratio. If Server A's weight is higher than Server B's weight, the policy assumes Server A's higher traffic handling capacity and maintains a longer queue of pending requests, compared to other Servers in the Pool. That is, the higher the server's weight, the greater the number of pending requests it will handle.

After getting the performance figures from the active peers (RTT or the number of pending requests), they are normalized between the value 1 and the maximal ratio (the default value is 100): The highest value is 1 while the lowest value is the max ratio value.

Queue Size Ratio policy is a dynamic algorithm and responds to external fluctuations upon request distribution.

Figure 18: Queue Size Ratio Policy




Internet
Clients
Router
SDC
SDC
Traffic is distributed between peers based on their real-time load, distributing to the peer with the highest availability.
1
2
3
4
5
Remote Peers

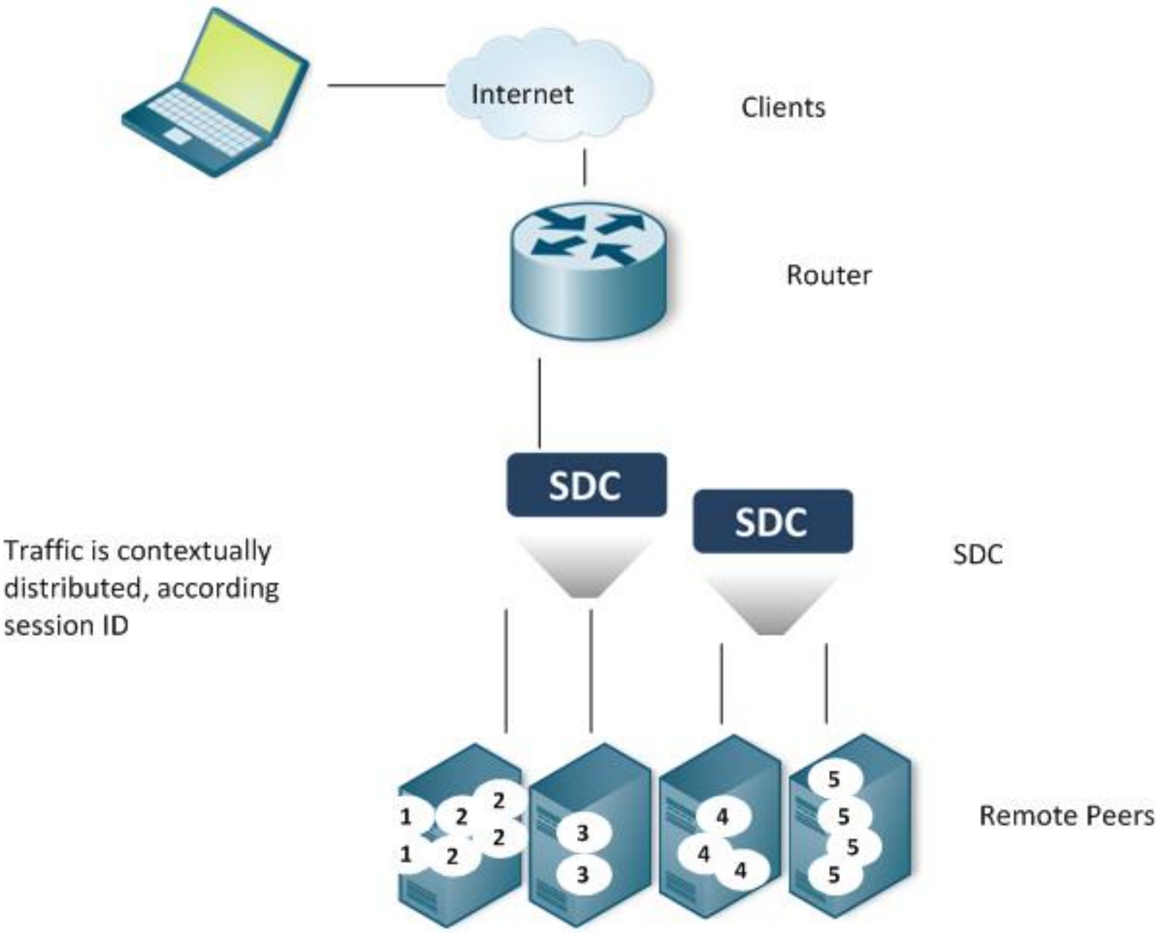
3.3.6.3.6 Contextual

When selecting a **Contextual** policy, load balancing policy maps the clients' session ID's to a list of available server peers. This way messages are sent to a specific server peer according to the session they belong to.

Note: You may set a different context-Id than the session ID using the groovy scripts. The setting is done by calling session.setContextId().

Messages sharing the same session ID will always be sent to the same server within a specific Session Timeout, regardless of the amount of messages handled within the session, and regardless of the SDC instance handling them, as shown in *Figure 20*.

Figure 19: Contextual Policy

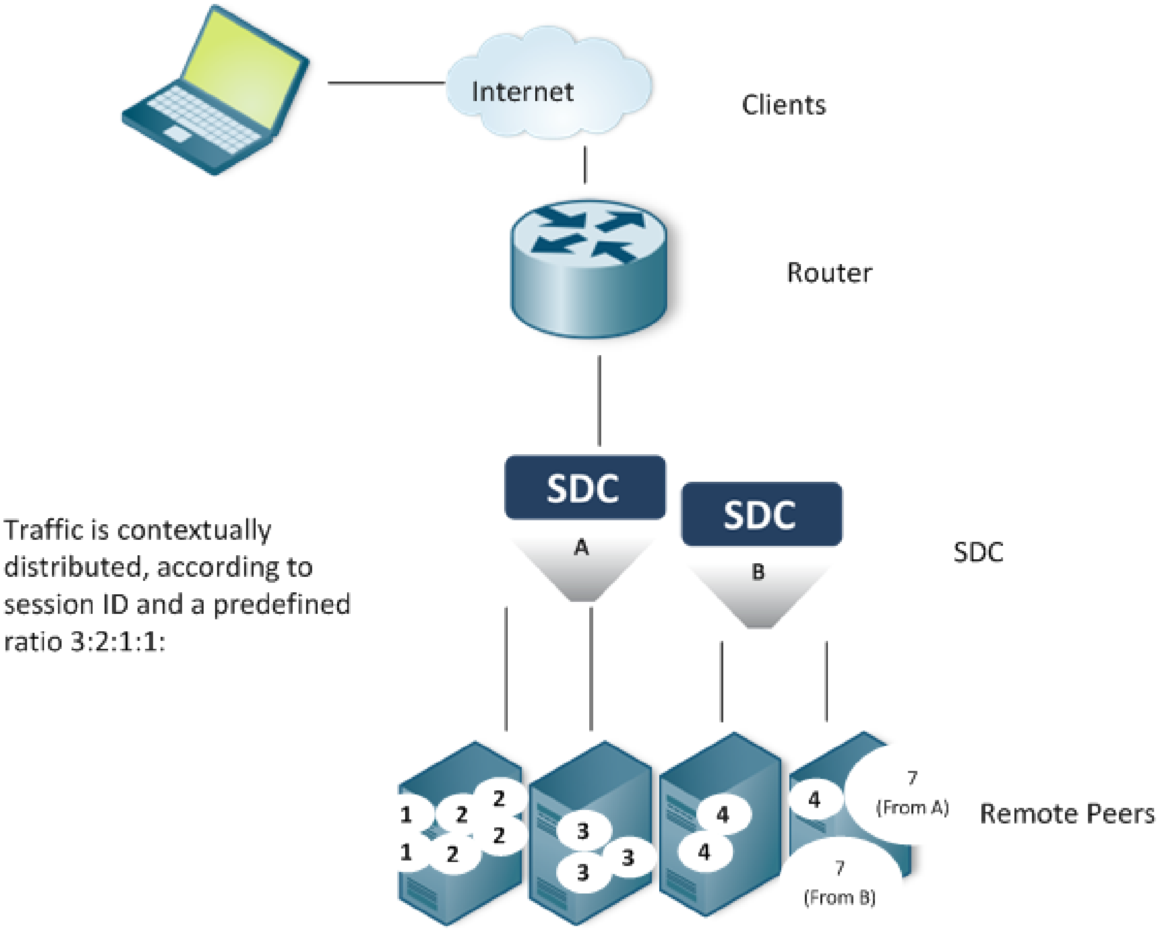


3.3.6.3.7 Weighted Contextual

When selecting a **Weighted Contextual** policy, the load balancing policy maps the clients' session ID's to a list of available server peers. This way messages are sent to a specific server peer according to the session they belong to. In addition to the session ID parameter, traffic distribution is also controlled by a predefined proportion. The weight of each server peer is set when establishing it and should be based upon its ability to handle incoming requests.

Note: Messages sharing the same session ID will always be sent to the same server within a specific Session Timeout, regardless of the amount of messages handled within the session, and regardless of the SDC instance handling them, as shown in *Figure 21*.

Figure 20: Weighted Contextual Policy

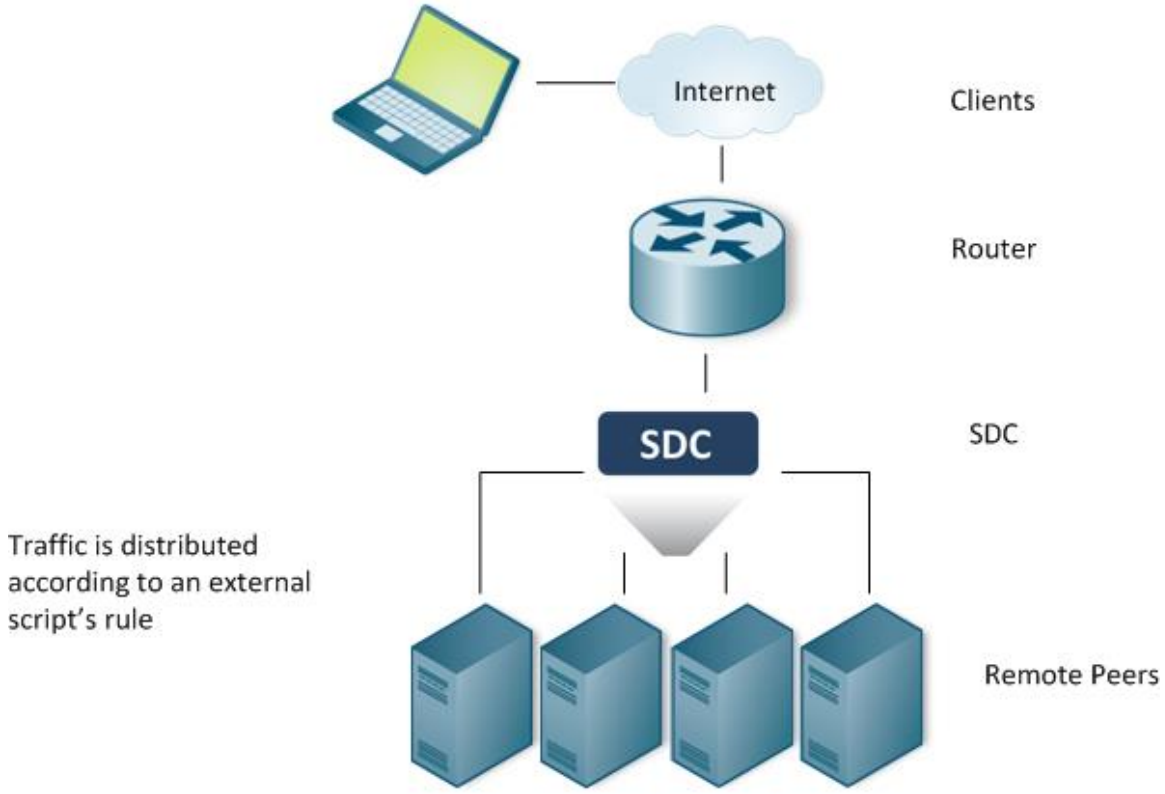

Internet
Clients
Router
SDC
A
SDC
B
SDC
Traffic is contextually distributed, according to session ID and a predefined ratio 3:2:1:1:
1
2
2
1
2
2
3
3
3
4
4
4
7
(From A)
7
(From B)
Remote Peers

3.3.6.3.8 External

When selecting an **External** Policy, the request's destination server peer is selected according to an external script's rule. External load balance policy may use a peer selector which its policy is set as a value of the Peer Selection script's argument (the policy may be used, for example, as a default policy when no server meets the specified script. This must be defined by the script).

Incoming requests are distributed as shown in *Figure 22*.

Figure 21: External Policy


Internet
Clients
Router
SDC
SDC
Traffic is distributed according to an external script's rule
Remote Peers

1. **To use an external script as the Policy's selection rule:**

2. From the **Policy** drop-down list, select **External**.

3. From **Internal Peer Selector Policy** drop-down list, select a policy that is used by the peer Selector argument in the Peer Selection script (the policy may be used, for example, as a default policy when no server meets the specified script. Using the peerSelector must be explicitly defined by the script, see example below).

In **Peer Selection**, type in the script according to which traffic is distributed across the available Remote Peers.

Table 21 details the parameters that SDC provides to the script:

Table 20: External Script Parameters

External Script’s Returned Value Type: Peer	
Parameter	**Type**
Request	Message
peerSelector	PeerSelector
peerTable	Peer Table
activePeerList	List<TransportPeer>
Session	Session
originPeer	Peer

Note: You may only call API methods associated with the parameters include in the above table. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

4. Click **OK**.

Example of External Script:

```
<ExternalSelectors>
     <ExternalSelector policyName="Hash" poolName="zone-b">
                                                <SelectionScript><![CDATA[
 /*
 * Looking for the peer in the UserTable,
 * If it is not in the pool table, using peerSelector
*/
      def peer = null;
      def key=session.getSessionId();
      if (key != null) {
      userTraceLogger.debug("looking for peer with key: " + key);

// getting the reference for the UserStorage
      def provider = UserStorageFactory.getProvider();

       def routingTable = provider.getUserTable("RoutingTable");
 // getting "peer Identity" (peer name)
        String peerIdentity = routingTable.get(key);
        userTraceLogger.debug("found for a key: " + key + " the following peer: " +
peerIdentity);
       if (peerIdentity != null) {
       userTraceLogger.debug("getting peer " + peerIdentity + " from peer table for
key:" + key + ", provider " + provider);
 // getting the "peer" object
       peer = peerTable.getPeer(peerIdentity);
                }
  // if the destination is not in the table, should add an option to decide that the
message is not routable, if destinations are not provisioned
     if (peer == null && activePeerList.size() > 0) {
// if the above was not found, using peerSelector, according to its policy
     peer = peerSelector.select(request, activePeerList, session, sourcePeer);
     userTraceLogger.debug("allocating peer " + peer.getName() +" for key:" + key +
", provider " + provider);
     routingTable.put(key, [peer.getName(), "zone-b", session.getSessionId()]);
   }
    } else {
    userTraceLogger.log(Level.WARN, "failed to lookup, Framed-IP-Address is missing
for " + request);
    }
return peer;
    ]]></SelectionScript>
       </ExternalSelector>
                </ExternalSelectors>
```

3.3.6.4 Assigning a Load Balancing Policy between Pools

In a deployment that has more than one pool, you can set a load balancing policy between pools to define how messages are routed to the different pools. The two options are **By Precedence** or **Round Robin**. The default is **By Precedence**, and messages are routed **to** only one pool, and only in the case that all the pool's peers are disconnected, that messages will then be routed to another pool. If you select a **Round Robin** policy, messages will be routed to all pools equally. Within the selected pool, the message is then routed to the peers based on the pool's configured load balancing policy (between peers). This option is configured per routing rule (**Flows** > **Flows** > **<Flow Name>** > **Routing Rules**>**Rule Configuration**>**Load Balancing Policy**), for more information see *Defining Rule Configuration Parameters*.

3.3.6.5 Assigning a Broadcast Pool Policy

Broadcast Pool policies are used when messages sent to a pool are routed to all of the server peers in the pool. This method of sending the same message request to all peers in a pool requires the user to configure an answer policy that defines how to handle the multiple answers from the different peers.

To assign a Broadcast Pool policy:

1. Go to **Topology** > **Pools**.
2. Click **Edit/Add** to edit/add a pool.
3. From the **General** tab, select **Broadcast Pool**.
4. Select one of the following answer policies from the drop-down list.

- **Don't Wait** – SDC does not wait for an answer and a Remote Node Event ("NO_REMOTE_EVENT") is sent. All received answers are discarded and the answer is created locally in the Handle Server Error script.

Note: The user must add a Handle Server Error script so that it can be invoked when the **Don't Wait** answer policy is selected.

The following is an example of a Handle Server Error script:

```
If (event)++RemoteNodeEvent.NO_REMOTE_EVENT)     {
def answer = requestFromClient.createAnswer (2001) ;
Answer.setError (false) ;
Return answer ;
}
```

- **Wait for Any (First)** – The first answer, regardless from which peer, to be returned is forwarded, and all others are discarded.
- **Wait for All - Conditional** – The first answer that matches the conditions that were set in the Check Error in Answer script are forwarded, all others are discarded.

Note: The user must add a Check Error in Answer script so that it can be invoked when the **Wait for All - Conditional** answer policy is selected.

The following is an example of a Check Error in Answer script:

```
If  (answer.getValue("Origin-State-Id") ==3) {
Return  RemoteNodeEvent.OK
}
Return  RemoteNodeEvent.NO_REMOTE_EVENT
```

Click **Save**.

5.

3.3.6.6 Configuring a Notification Pool

You can configure requests to be sent to another server, in addition to the configured destination server. The group of additional servers are configured as part of a notification pool and requests can be copied to it. Use cases for this feature could be for saving user data statistics information (to the notification pool) separate from billing or roaming management (handled by the "regular" destination pool).

A notification pool can be configured with a load balancing or broadcast policy. If configured as a Broadcast Pool, all answers are discarded, as sending back answers is not relevant for notification pools.

During routing, requests configured with a routing rule that have been assigned to a notification pool will be sent to one of the selected configured notification pools (in addition to the "regular" destination pool), according to the configured notification policy that defines when the request is sent to the notification pool.

Configuring a Notification Pool includes the following procedures:

- Configuring a pool as a notification pool
- Assigning a routing rule with pool notification to a request

Note: Requests sent to a notification pool are not counted as sent messages that are calculated for the “Licensed MPS” KPI. All other pool statistics and traps are relevant to notification pools (for more information, see cross ref).

Guidelines and Restrictions

Before configuring the routing decision in routing table.

Routing Action	Destination	Action done
ROUTE	1. Pool 2. Notification Pool	The routing decision will select a peer from a pool in both: main pool and notification pool (based on the load balancing method which is configured in each pool)
FORWARD	1. Peer 2. Notification Pool	The routing decision will get the selected peer name and will send the message only to the peer and NOT to the notification pool
FORWARD	1. Peer 2. Broadcast Pool	When using a broadcast pool, the message is sent to all peers and no peer selection is needed. Therefore, the routing decision will get the selected peer name and broadcast the message to the broadcast pool
FORWARD	1. Pool 2. Notification Pool	The routing decision will apply the peer selection from pool (same method used in "ROUTE") and the message will be sent to a peer in the destination pool as well as in the notification pool

When configuring message requests to be copied to a notification pool, you must first configure at least one pool as a notification pool and then assign a message's routing rule to a notification pool.

To configure a pool as a notification pool:

1. Go to **Topology>Pools>Add Pool**, select **Notification Pool**.

   The pool is now configured to be a notification pool. The **Message Prioritization** and **Ramp Up** configurations are grayed out, as they are not relevant for notification pools.

Note: Once a pool is selected as a Notification Pool, you cannot edit this configuration. To change the configuration, delete the pool and then add a new pool.
A Notification Pool can be configured with a load balancing or broadcast policy. The behavior of a Notification - Broadcast Pool is that all messages are discarded (**Don't Wait**).

To assign a routing rule to so that a message is copied to a notification pool:

1. Go to **Flows > Flows > <Flow Name> > Routing Rules**.
2. Select a Routing Rule and select **Rule Configuration**.
3. Select **Assign Notification Pool**.

   The Notification Pool box opens.

4. In the **Select Notification Pool** dropdown, select one of the configured notification pools.
5. Select a Notification Policy to define when a request will be sent to the selected notification pool:

   ▪ **Upon Request** – the request will be sent to the notification pool when it is sent to the "regular" destination pool

- **Upon Answer** – the request will be sent to the notification pool only after an answer (any or only a successful answer) from the "regular" destination pool is received, as selected from the dropdown:
    - **ANY_ANSWER**
    - **SUCCESSFUL_ANSWER**

6. Under **Max Resend Attempts**, set the maximum number of request sending retries, in case request fails.

7. Under **Delay Between Attempts (Millis)**, set the time difference between one resend attempt and another.

To configure a pre-routing transformation script for a request to be sent to a notification pool:

1. Go to **Flows > Flows > <Flow Name> > Transformation>Pre-Routing**

2. Under **Script**: include the envelope parameter is_notification_event.

3.3.6.7 Configuring Pool Failover Policy

When one of the pools is out of service, the SDC reroutes messages belonging to an existing session to a peer in the next available pool. When configuring the routing rule for a session, you have the option to disable this behavior. When disabled, messages belonging to an existing session are not rerouted to a peer in a different pool when the pool selected for the session is out of service.

1. The pool failover policy is configured per routing rule.

2. **To configure a pool failover policy:**

Go to **Flows > Flows > <Flow Name> > Routing Rules>Rule Configuration**

Select the **Route Ongoing Session Messages to Selected Pool Only** checkbox so that messages matching the routing row will not to be routed to peers in another pool when the original destination pool is out of service.

Note: When there are multiple pools in an SDC site and one pool is out of service, proxied messages will not be sent to a pool that contains a geo-redundant peer.

3.3.7 Configuring Peer Failover Policy

You can decide if you want messages within the same session to always be routed to the same destination peer even when that destination peer is not available.

To configure a peer failover policy:

Go to **Topology>Pools>Add/Edit>General>Pool Usage**.

1. Select **Route Ongoing Session Messages to the Selected Peer Only** if you do not
2. want peer failover.

Note: The checkbox is not selected, by default, meaning peer failover is enabled.

3.3.8 Configuring Overload Control Policy

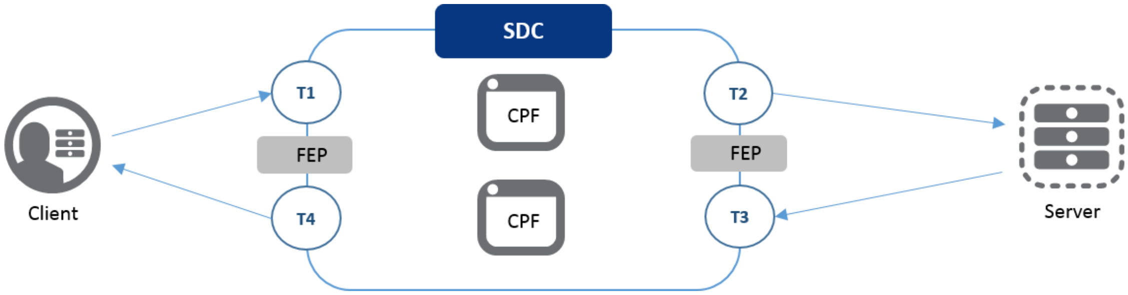
The SDC protects peers and pools so that they do not become overloaded. This is done by applying rate limits, thresholds, and ramp-up times. Once a peer is in an overloaded state, the SDC offers different ways to configure how messages will be processed.

The Answer Policy configuration is configured on the peer level and defines the behavior of the SDC once an outgoing peer rate limit has been exceeded. The Message Prioritization mechanism, which is configured on the pool level, optimizes request processing between the peers in a pool in spite of one or more of the peers being in an overloaded state. Ramp-up policy helps prevent peer or pool overload by limiting the message traffic to the peer or pool during initialization.

3.3.8.1 Configuring Rate Limits

Rate limits are configured to monitor and control the amount of traffic that the SDC receives from either a client or server peer and/or sends toward a peer or a pool of peers. These limits are configured by the number of messages and/or bytes that the SDC can receive and/or send. The basic traffic flow between the SDC and the networks is illustrated below.

Figure 22: Basic Traffic Flow between the SDC and Networks



In this flow, message requests are sent from clients, received by the SDC, and then sent by the SDC to a server. Message answers are then sent from the server back to the SDC, and then sent by the SDC to the client.

This flow includes two types of traffic– incoming (from the client/server to the SDC) and outgoing (from the SDC to the client/server). The volume of traffic received by the SDC at an entry point (T1, T3) or exit point (T2, T4) is monitored based on configured limits. These limits ensure that the overall traffic flow performance is constantly monitored and when handled to help ensure that there is no service degradation in overload conditions.

The peer rate limit is configured in a peer profile and then applied to a peer as part of applying a peer profile. In addition, you can configure the related alarm thresholds (critical, major, minor) for outgoing and incoming rate limits by peer (see *Threshold Management*).) The pool rate limit is configured as part of the pool configurations when adding or editing a pool. For a pool, you can configure the related alarm thresholds (critical, major, minor) for outgoing TPS (see *Threshold Management*).

The FEP and CPF are automatically configured with an internal rate limit during installation. The FEP rate limit is dynamic as it adjusts its rate limit by 30K TPS each time it connects or disconnects with a new CPF. The CPF rate limit is 20K TPS for virtual machine deployments and 22K TPS for bare metal deployments. Once these rate limits are reached, traffic is not processed.

3.3.8.1.1 Configuring the Incoming Traffic Rate Limits for Peers

The incoming traffic rate limits are configured to monitor the amount of traffic that the SDC receives from either a client or server peer. These limits are configured by the number of messages and/or bytes that the SDC can receive. The incoming rate limit is used to calculate the Current Incoming TPS vs Peer Rate Limit alarm thresholds (see *Configuring Alarm Thresholds*).

To configure the Incoming Traffic Rate Limit per Peer:

Go to **Topology>Peer Profiles>Add/Edit>Rate Limit**.

1. In **Incoming Rate Limit**.

2.

a. In **Incoming Requests per second**, set the limit of messages that can be received per second from the peer.

b. In **Incoming Bytes per second**, set the limit of bytes that can be received per second from the peer.

3. Click **Save**.

3.3.8.1.2 Configuring the Outgoing Traffic Rate Limits for Peers

The outgoing traffic rate limit is configured to monitor the amount of traffic that the SDC sends to either a client or server peer. The outgoing rate limit is used to calculate the Current Outgoing TPS vs Peer Rate Limit alarm threshold (see *Configuring Alarm Thresholds*).

In addition, when a peer exceeds its configured outgoing peer rate limit, the peer is in an overloaded state, which triggers the SDC to handle traffic according to the configured
1. Answer Policy (see *Configuring Answer Policy*.)

2. **To configure the Outgoing Traffic Rate Limit per Peer:**

Go to **Topology** > **Peer Profiles** > **Add/Edit>Rate Limit**.

In **Outgoing Rate Limit**.

In **Outgoing Requests per second**, set the limit of messages that can be sent per second to the server peer.

Click **Save**.

a.

3.3.8.1.3 Configuring the Outgoing Traffic Rate Limits for Pools

3. The outgoing traffic rate limit for pools is configured to monitor the amount of traffic that the SDC sends to a pool. The outgoing pool rate limit is used to calculate the Current TPS vs. Pool Rate Limit alarm thresholds (see *Configuring Alarm Thresholds*).

To configure the Outgoing Traffic Rate Limit per Pool:

Go to **Add Pool** > **General**.

1.

In **Pool Rate Limit (MPS)** field, enter the desired rate limit.

2.

Click **Save**.

3.

3.3.8.1.4 Configuring Alarm Thresholds

The SDC offers configurable thresholds for incoming and outgoing traffic monitoring. These thresholds are based on the configured rate limits for peers and pools and allow early detection of potential peer and pool overloads.

Upon reaching the defined thresholds, per pool and/or peer, the system sends an SNMP trap (Pool Rate Limit State Change and/or Peer Rate Limit State Change) indicating that the measured TPS has exceeded one of the configured peer and/or pool thresholds (Minor, Major, Critical). These thresholds can be configured globally (see 🌐 *Threshold*
1. *Management*) or per peer or per pool.

2. To configure the alarm severity thresholds for overloaded peers:

Go to **Topology>Peers>Add/Edit>Thresholds**

In **Incoming TPS vs Rate Limit**, set the threshold percentages for generating rate limit alarms by severity level (**Critical**, **Major**, **Minor)** for when the number of requests sent from a peer to the SDC has exceeded its (peer profile configured) incoming rate limit.

3. In **Outgoing TPS vs Rate Limit**, set the threshold percentages for generating rate limit alarms by severity level (**Critical**, **Major**, **Minor)** for when the number of requests sent to a peer from the SDC has exceeded its (peer profile configured) outgoing rate limit.

4. Click **Save**.

To configure the alarm severity thresholds for overloaded pools:

1. Go to **Topology>Pools>Add/Edit>Thresholds**

2. In **Outgoing TPS vs Rate Limit**, set the threshold percentages for generating rate limit alarms by severity level (**Critical**, **Major**, **Minor)** for when the number of requests sent to a pool from the SDC has exceeded its pool configured rate limit.

3. Click **Save**.

3.3.8.1.5 Configuring Answer Policy

Once the outgoing peer rate limit has been exceeded, the peer is in an overloaded state. You can configure how the overloaded peer will respond to requests sent from the SDC.

To configure the Answer Policy for an overloaded peer:

1. Go to **Topology>Peer Profiles>Add/Edit>General**

2. Under **Overload Answer Policy**, select one of the following:

- **Silent Discard** – discards all incoming messages and does not return any answer.

Note: This is the default behavior.

- **Return Busy Answer** –returns the request to the SDC and a BUSY Remote Node Event is sent

3. Click **Save**.

3.3.8.2 Configuring Message Prioritization

Message Prioritization optimizes request processing in overloaded peers within a pool. When Message Prioritization is configured for a pool, the SDC processes requests based on their priority, with high priority being processed first, then medium, and then low.

An overloaded state for message prioritization is determined if any of a pool's peers' threshold KPIs for timeout request, busy error answer, and round trip time has exceeded their configured threshold levels. These peer thresholds are configured by peer profile. For more information about configuring thresholds, see *Configuring Peer Profiles*.

Once one of these thresholds is passed, Message Prioritization is initiated, assuming that the overloaded peer belongs to a pool that is enabled with Message Prioritization. As part of Message Prioritization, there is a watermark level for the overloaded peer. The watermark is a dynamic indicator of a peer's current capacity and it is calculated every 0.5 seconds. When one of the three thresholds is exceeded, the peer's watermark capacity is dropped by 20 percent of the TPS, thereby limiting the number of prioritized messages that will be sent to the overloaded peer. The SDC reviews the last second of message processing to see if the peer was able to process all of its messages. High messages will take priority over Medium messages which take priority over Low messages. The SDC is configured to consider a priority level only if all messages for that priority level can be processed by the peer.

When Message Prioritization is not enabled, the SDC does not consider a peer's thresholds and no watermark capacity limit is calculated and messages are processed according to the selected load balancing policy, with the risk of any overloaded peers discarding high priority messages.

The SDC will attempt to route any prioritized messages that are not successfully sent to a peer within a pool to another available pool associated with the routing rule. Once the peer returns to below its threshold limit, the SDC will stop prioritizing messages.

When the number of peers in the pool that are available to process low priority messages is lower than the defined minimum number of active peers configured for the pool, the pool’s state is partially out of service and in **Topology > Pools > State**, the pool will show as Limited. When the number of peers in a pool that can accept, any priority is lower than the defined number of active peers configured for the pool, the pool’s state is out of service and in **Topology > Pools > State**, the pool will show as Limited or Closed, depending if it is out of service for some or all of the CPFs.

Note: When rate limit and Message Prioritization are active at the same time, messages routed to an overloaded peer are first assessed based on the configured rate limit and overload control policy, regardless of how messages were prioritized. Only those messages that are above the configured rate limit will be processed according to message prioritization.

Message Prioritization includes the following procedures:

- Configuring Message Prioritization Thresholds
- Configuring a Message Prioritization Rule
- Enabling a Pool with Message Prioritization

Note: This feature assumes that the peers in the pool have been configured with defined timeout request thresholds, busy error answer thresholds, and round trip time thresholds in the peer profile configuration. Message prioritization starts approximately 1000 milliseconds after a peer has reached its overloaded state.

3.3.8.2.1 Configuring Message Prioritization Thresholds

A peer's overloaded state is detected by three thresholds that are configured per peer profile: Timeout Request, Busy Error Answers, and Round Trip Time. When one of these thresholds, which are KPIs, have been crossed/exceeded, the peer is considered partially out of service and in an overloaded state. Part of being defined as an overloaded peer means that Message Prioritization is triggered if the overloaded peer is part of a pool that is enabled with Message Prioritization.

To configure the Message Prioritization Thresholds:

1. Go to **Topology>Peer Profiles>Add/Edit>General**.

2. In **Timeout Threshold,** set the time frame (in milliseconds) in which the peer is expected to answer requests.

Note: When configured in routing (**Flows** > **Flows** > **Routing>Rule Configuration** > **Max Resend Attempts**), the request is resent if the defined timeout expires before the peer sends an answer. The default is set to three seconds. Timed-out requests are also counted for determining a server peer's health. For additional information on Health Monitoring, see *Health Monitoring*.

3. In **Timeout Threshold**, set the allowed ratio between the number of requests sent to the peer and the number of requests not answered by the peer in the defined timeout period.

Note: This indicator is used for determining a server peer's health and for triggering message prioritization overload control.

4. In **Error Answers Threshold**, set the allowed ratio between the number of requests sent to the peer and the number of error answers returned from the peer.

Note: This indicator is used for determining a server peer's health.

5. In **Busy Error Answers Threshold**, set the allowed ratio between the number of requests sent to the peer and the number of error answers (with DIAMETER_TOO_BUSY result code) returned from the peer.

6. In **Round Trip Time Threshold** set the threshold for the allowed round trip time.

Note: This indicator is used for determining a server peer's health and for triggering message prioritization overload control.

Click **Save**.

3.3.8.2.2 Configuring a Message Prioritization Rule

7. You need to create a Message Prioritization Rule with a defined priority level so that the SDC will know how to handle messages that are sent to a pool with peers in an overloaded state.

A Message Prioritization rule contains one or more Traffic Prioritization Attributes (message AVPs), and a priority (high, medium, or low). When a request's AVP matches the Message Prioritization Rule attributes, the request is assigned the corresponding priority. Once a Message Prioritization Rule has been configured, you can enable overloaded pools in your environment to prioritize messages.

To configure a Message Prioritization Rule:

1. Go to **Administration>Message Prioritization**.

2. Select **Add**.

A new rule (MP) is added with a default Priority level of MEDIUM.

3.

4. To add attributes to the message prioritization rule, click **Rule Attributes**. For information about adding **Rule Attributes**, see *Adding Rule Attributes*.

In the **Priority** column, select HIGH, MEDIUM, or LOW from the drop-down list.

5. Note: If no matching rule is found, the default priority, (MEDIUM), is used.

Click **Submit**.

3.3.8.2.3 Enabling a Pool with Message Prioritization

1. Message Prioritization optimizes request processing in overloaded peers within a pool.

To enable a pool so that messages are prioritized when a pool's peers are in an overloaded state:

Go to **Topology** > **Pools** > **Add>General**.

Select **Activate message prioritization upon approaching overload control criteria**.

Click **Save**.

2.

3.3.8.2.4 Configuring Peer Ramp Up

3. The Peer Ramp Up mechanism prevents a specific peer from being in an overloaded state during startup or after being out of service, busy, or partially out of service. During the ramp up period, which lasts a minimum of five seconds, traffic is sent to the peer on a gradual basis.

To configure a peer's ramp-up time:

Go to **Topology>Peers>Add/Edit>General.**

1.

2. In **Peer Ramp up (sec)**, enter the time (in seconds), that the peer will be in ramp-up mode from when the mode is activated.

3. Click **Save**.

3.3.8.3 Configuring Pool Ramp-up

The Pool Ramp Up mechanism prevents a specific pool from being in an overloaded state during startup or after being out of service, busy, or partially out of service. During the ramp up period, which lasts a minimum of five seconds, traffic is sent to the pool on a gradual basis.

1.

2. To configure a pool's ramp-up time:

Go to **Topology > Pools** > **Add>General**.

3.

In **Pool Ramp-Up Time (Seconds)**, enter the time (in seconds), that the pool will be in ramp-up mode from when the mode is activated.

Click **Save**.

3.3.8.4 Editing a Pool

This section describes how to edit a pool.

To edit a Pool:

Select a pool from the list and click **Edit**. The Edit Pool dialog box is displayed:

You can edit the enabled fields, as detailed in *Adding a New Pool*.

1.
3.3.8.5 Removing a Pool

2. You can remove any pool from a site.

To remove a pool:

Go to **Topology** > **Pools**.

1. Select the Pool from the table.

2. Click **Remove**.

3. A confirmation message appears.

4. Click **OK**.

3.3.9 Configuring the Access Control List

The Access Control List allows you to compose rules that determine which Client Peers are accepted by SDC and which are rejected by it. Client Peers are identified by their IP address and a matching Peer Profile. Accepted Client Peers may send requests to a Server Peers, while a rejected Client Peers may not do so.

When a Remote Client Peer tries to connect to SDC, its IP address is compared against the list of IP addresses of the ACL rules indicating an “Accept” action. If no rule’s address matches the Client, it is rejected (unless **Accept Unknown Peers** is selected). If a matching IP address is found, SDC waits for a CER (capabilities exchange request) and upon its arrival, compares the requesting client’s properties (IP address and the CER content) with the IP addresses and the Peer Profiles of all ACL rules. If a matching IP address and Peer Profile are found and the rule’s action is ‘Accept’, the capabilities exchange begins, otherwise the client is rejected.

Note: The ACL configuration, unlike IPTABLES configuration, does not affect existing connections.

To change the Access Control List:

Go to **Topology** > **Access Control List**.

To add a rule:

1. Click **Add** to add a new Client Peer rule to the list.

1. Under **Address**, enter the IP address of the Client Peer. A CIDR formatted address
2. may be entered, indicating range of IP addresses.

Note: CIDR (Classless Inter-Domain Routing) is the routing system used to allocate internet addresses more flexibly than the IP address allocation method, and thus creates a bigger range of addresses than the IP method (e.g. – 192.168.10.0/27).

3. Under **Peer Profile**, you can select a Peer Profile that the rejected or accepted Peer must match.

Note: ACL rules apply to client peers that are of the specified IP address and
4. match the selected Peer Profile.

5. Under **Action**, select whether to **Accept** or **Reject** the Client Peer.

1. Under **Enabled**, select whether this rule is enabled (**True**) or disabled (**False**).

2. **To change the order of the rules:**

Select the rule from the list

Change the rule's location in the list by clicking **Up** or **Down**.

Note: The Client Peers are checked against the rules in the list according to the order they are listed in. When a matching rule is found the rule examination is terminated.

To remove a rule from the list:

Select the rule from the list.

Click **Remove**.

1.

2. **To configure the default behavior in case no rule matches the connecting client IP:**

Select or clear **Accept unknown Client Peers**.

1. **To allow unknown Client Peers (Peers which do not appear in the list) to connect to SDC:**

1. Select **Accept unknown Client Peers**.

To reject these Client Peers:

1.

Clear **Accept unknown Client Peers**.

3.3.10 Health Monitoring

In the ongoing effort for creating highly available, scalable, reliable and resilient signaling plane, SDC supports Server Remote Peer health monitoring, used to verify that the back-end systems are operational and can handle incoming traffic.

A health monitor is generally set to test a specific parameter of a Server Remote Peer for an expected behavior in a predefined time frame. There are various types of health monitors, but in all cases, when the monitor's test indicates entity unavailability, you may stop routing traffic to it. The following categories can reflect a peer's status and are displayed in the peer table:

- Close (Out of service)
- Out of Service Partially

- Open (In service)

Health Monitors operate continuously to determine the availability of client and server remote peers. When a server remote peer becomes available again it is gradually directed with traffic.

SDC provides two types of Server Peer monitors:

- Error detection
- Proactive Service checking

3.3.11 Error Detection Monitor

This monitor tests the Server Peers' responsiveness to requests by checking if the number of errors in a predefined measuring interval exceeds a certain threshold. There are two types of error detection monitors:

- Timeout Monitor
- Response Analysis Monitor

The monitor is triggered upon each timeout event and for each received response.

3.3.11.1 Timeout Monitor

When SDC sends a request to a Server Peer and does not receive a response in an acceptable predefined time frame, it adds a "timeout" error to the accumulated number of "time out" errors received from that Server Peer.

3.3.11.2 Response Based Monitor

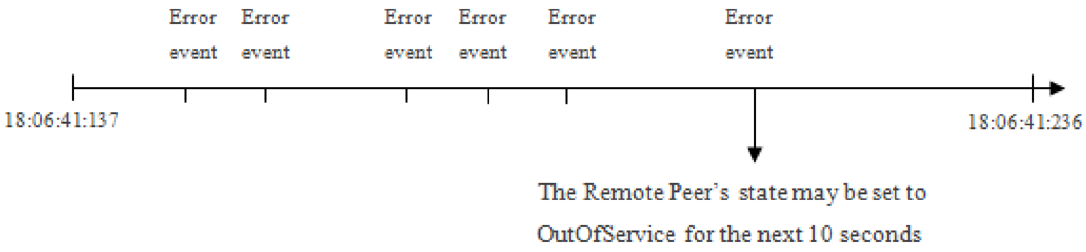
When SDC sends a request to a Server Remote Peer and receives a response that is considered an error, it adds the spotted error to the accumulated number of errors received from that Server Remote Peer.

Answer error detection is flexible. The SDC administrator may diagnose specific error cases (for example – a specific result code may indicate an error). Answer error detection is done by implementing the **Check Error in Answer Routing** script.

Note: For additional information on Routing scripts, see *Defining Routing Scripts*.

The number of errors is accumulated and you may decide, according a certain threshold in a specified time frame (for example, 6 error events in 100 Millis), to set a peer's state to Out of Service) as shown in *Figure 24*.

Figure 23: Error Events in a Measuring Interval



3.3.11.3 Setting an Error Detection Monitor Parameters

Error Detection Monitors are set per each Remote Peer by setting the error detection parameters.

3.3.11.4 Custom Service Availability Monitor

SDC's provides the ability to add custom and proactive service monitoring mechanism that can perform a wide range of tests: from simple tests, such as pinging each Server Remote Peer, to more sophisticated tests, such as assuring Server Peers are able to serve specific requests. It is possible to have multiple monitors perform any test that is required in order to assure service availability. Like other parts, health monitoring tests are configured and customized via script language. These health monitoring tests are performed in addition to other SDC’s tests when it attempts to send requests to Remote Peers and analyze responses from them.

3.3.11.5 Adding a Service Availability Health Monitor

Each service availability health monitor is implemented in a separate script. No limitation applies to the number of scripts, thus no limitation applies to the number of service checking procedures. Three elements comprise each service checking health monitoring script:

- Condition – the condition script which indicates whether the Remote Peer’s status should be checked using this specific script or not.
- Monitor Check – the health monitoring script.
- Interval (Millis) – long. The interval between the script executions.

Note: You may only call API methods associated with the Health Monitor parameters. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

A monitor runs on a recurrent basis, the recurrence is controlled by this interval. Each of the Remote Peers is examined to determine whether or not it matches the condition’s criteria. When a Remote Peer matches the condition’s criteria, a monitor check script is run. The script examines the Remote Peer’s check result and you may decide, according to the check result, whether to set the Peer’s state to “Out of Service,” “Out of Service Partially”, or alternatively, set its state to “Back to Service”. When the Remote Peer state is “Out of Service” no further requests are delivered to it, until its state is set back to “Back to Server”. A peer in an “Out of Service Partially” state will process existing sessions while not accepting new sessions.

```
def roundtrip = peer. getRoundTripTimeMillis ();
if(roundtrip >= 200){
peer.outOfService(5, java.util.concurrent.TimeUnit.SECONDS);
}
else {
peer.backToService();
}
```

To set SDC's Remote Peer Service Checking Availability Health Monitor:

1. Go to **Topology** > **Health Monitoring** > **First Type**.

2. In **Condition**, type in the condition's script which indicates whether the Remote Peer's status should be checked using this specific health monitoring script or not.

Note: The Service Checking Health Monitor condition script typically includes verifying that the Remote Peer is part of a group of peers which should be tested by the Monitor Check script with the specified script.

Table 22 details the Health Monitoring Condition Script parameters.

Table 21: Health Monitor Condition Script Parameters

Health Monitor Condition Script's Returned Value Type: Boolean	
Parameter	**Type**
Peer	Peer
userTraceLogger	UserTraceLoggerWrapper
Metadata	MetaData

Note: You may only call API methods associated with the parameters listed in the above table (to view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

3. In **Monitor Check**, type in the health monitoring script.

Note: When a peer is defined as partially out of service in a script, and it is an overloaded state, the peer may accept new sessions in addition to existing sessions.

Table 23 details the Health Monitoring Check script parameters.

Table 22: Health Monitor Check Script Parameters

Health Monitor Check Script's Returned Value Type: none	
Parameter	**Type**
Peer	Peer
userTraceLogger	UserTraceLoggerWrapper
metaData	MetaData

Note: You may only call API methods associated with the parameters listed in the above table (to view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

4. In **Interval (Millis)**, type in an interval (in milliseconds) defining the time between monitor checks.

Note: The minimum interval value is 1000 milliseconds.

5. Click **Submit.**

3.3.12 Topology Architecture

The following section describes the SDC Topology architecture.

Figure 24: SDC Network Topology

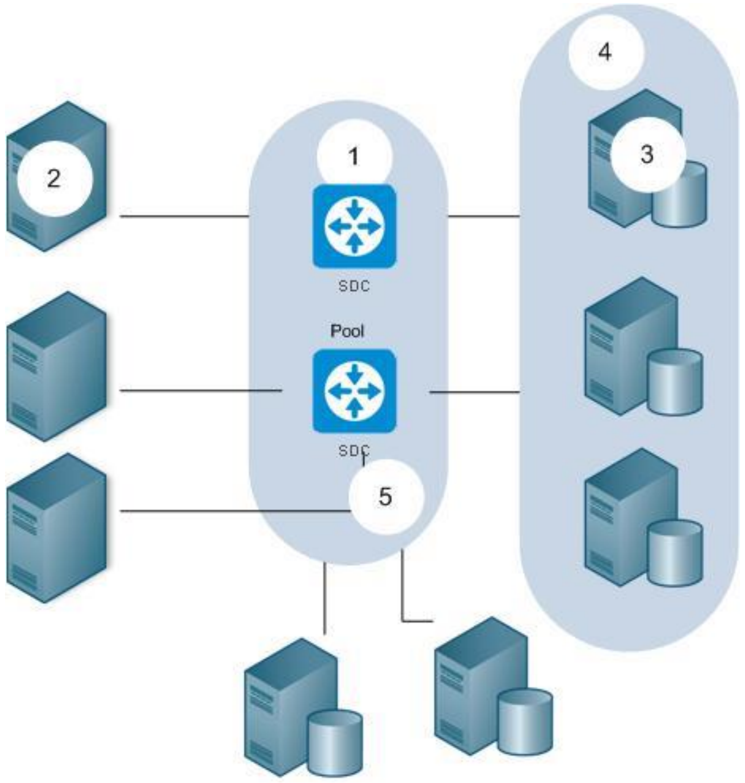


Table 23: SDC Network Topology Legend

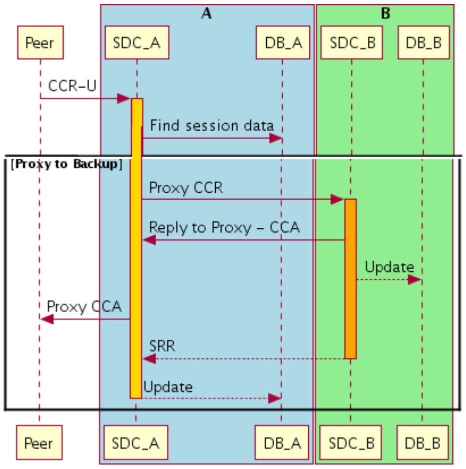
Number	Topology Object	Description
1.	SDC	An instance of SDC in the Cluster (CPF + FEP).
2.	Client Peer	A client node in the NGN network that consumes AAA services.
3.	Server Peers	A server node in the NGN network that provides AAA services.
4.	Pool	A group of peers
5.	Cluster	A group of SDCs used to provide translation and connectivity services and support high availability.

3.4 Proxying Requests between SDC Sites

An SDC peer, which receives a request, may handle the request or proxy the request to a remote site. Proxying the request is performed when the session is unknown to the local site or session binding fails and the remote site has the required data to handle the incoming request. Proxying is supported for Diameter and HTTP peers and is enabled by configuring the FEP at the geo redundant site as a Diameter/HTTP peer and selecting the **Use for Geo Redundant Sites Connection** checkbox (**Topology>Peers>Add/Edit>General**).

Note: For a pool to be geo-redundant, at least one of its peers must be configured with the **Use for Geo Redundant Sites Connection.**

Figure 25: Request Proxying



3.5 Replicating Session Data

By saving (persisting) session data (i.e. Session ID, Destination, session stickiness) and Binding Keys in a repository, SDC can then query future incoming requests to see if there is a relevant existing session that meets the defined criteria, thereby allowing the request to be consistently routed to its destination peer.

Session data is saved in data tables in the Session Repository once the session destination is determined. All session data is replicated and synchronized in Session Repository instances either within an SDC site or in Session Repository instances located on mated SDC sites, depending on your deployment. Whether or not a session is saved and where it is saved depends on the **Session Persistence Policy** (**Persist**, **Non Persistence**, **Persist**

and Replicate) that is configured per session. For more information on configuring a persistency policy, see *Session* Management.

3.6 Default Transport Configuration

Default Transport Configuration is a collection of default parameters which control the behavior of TCP and SCTP channels. The socket defaults affect the way each Remote Peer is treated by SDC in case the Peer was not individually configured.

Note: Default Transport Configuration parameters may be applied individually per Peer Profile and globally, per SDC. In case unknown Peers are connected to SDC, the values are applied to it.

For parameters, such as Receive/Send Buffer Size (Server & Client) that are applied during the initial connection (INIT) of dynamic peers with the SDC, they are not configurable per dynamic peer profiles as they assume the configured or default (OS) values of static peer profiles. Dynamic transport (SCTP or TCP) peer profiles are associated with a specific virtual server. To configure the Receive/Send Buffer Size (Server & Client) parameters, from the virtual server go to **Administration > Specific Site Settings > Default Transport Configuration > SDC** component drop-down list > select the relevant FEP component (virtual server).

1. **To change the Default Transport Configuration:**

2. Go to **Administration > Specific Site Settings > Default Transport Configuration**. The Default Transport Configuration screen displays the **Socket Defaults** tab.

   From the **SDC component** drop-down list, select the **CPF/FEP** Node that you want to apply the Default Transport Configuration changes.

Table 24 details the Socket Defaults and their descriptions:

Note: To use the values from the operating system configuration, select the OS option.

The displayed Default Transport Configuration parameters reflect the last configured channel parameters. Once a peer connection is made, these parameters cannot be edited, unless the peer connection is reset by disabling and then enabling the client and server peers.

You can view the current configured parameters for SCTP and TCP connections in the FEP logs with the following API request that can be called from the Groovy scripts:

```
 if (stack.isProxy()){
     ((com.traffix.openblox.core.transport.connection.TransportPeer)
stack.getPeerTable().getPeer("server1")).getTransportWrapper().printSocketOptions();
    }
```

Note: Asymmetrical routing is enabled by default. Traffic between a client/server (such as MME) to the SDC is automatically cross pathed to a second IP address, in the event that the initial path to the first IP address fails. To change this default behavior, contact *F5 Support*.

Table 24: Socket Defaults

Parameter	Description
Buffers (TCP and SCTP)	
Send Buffer Size (Server & Client)	The TCP and SCTP sending buffer size (in bytes for outgoing data).
Receive Buffer Size (Server & Client)	The TCP and SCTP receiving buffer size used (in bytes for incoming data).
Socket Options (TCP and SCTP)	
TCP No Delay (Server & Client)	Disable Nagle's algorithm for this connection. Written data to the network is not buffering pending acknowledgement of previously written data.

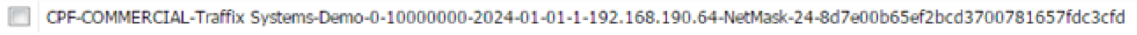
Parameter	Description
	Note: This parameter is set to true by default to minimize latency issues.
So Linger (Server & Client)	Specifies the timeout (in seconds) for brute-force shutdown of a channel, after a close request (TCP level) is sent from SDC to a remote node.
Reuse Address (Server)	When enabled, used for MulticastSockets in java, and it is set by default to True for MulticastSockets
TCP	
Keep Alive (Server)	When enabled and no data has been exchanged across the socket for two hours*, TCP automatically probes the Remote Peer. One of following responses is expected: ▪ ACK – no error occurred. The application is not notified and TCP sends another probe following another two hours of inactivity. ▪ RST - the Peer's host has crashed and rebooted. The socket is closed. ▪ No response. The socket is closed. Note: The period may be configured per SDC.
Traffic Class (Server)	This option sets the type-of-service or traffic class field in the IP header for a TCP or UDP socket.
SCTP	
Heartbeat Interval (Server)	This is the interval (in seconds) when a HEARTBEAT chunk is sent to a destination

Parameter	Description
	transport address to monitor the reachability of an idle destination transport address.
Cookie (Server)	Handle COOKIE PRESERVATIVE parameter (in milliseconds) in the INIT chunk.
Number of Inbound Streams (Server & Client)	The number of SCTP inbound streams.
Number of Outbound Streams (Server & Client)	The number of SCTP outbound streams.
Support Unordered Delivery (Server & Client)	Enable support for accepting and processing SCTP data chunks as they arrive, even if they are out of sequence.
SCTP Profiles	
SCTP Profile	The SDC contains the following preconfigured SCTP profiles. Each profile is configured with pre-defined parameters. Select one of the following profiles or select **Custom** to configure a unique SCTP profile: OS Same US State US Coast to Coast and Inside EU Asia-Asia EU-USA EU/US - Asia Universal
Association Max Retrans (Server & Client)	Maximum number of retransmission attempts to a peer per association, by message type.
Pa**th Max Retransmits (Server & Client)**	Maximum number of retransmission attempts to a peer per path, by message type.

Parameter	Description
RTO Initial (Server & Client)	The initial value of RTO (retransmission timeout in milliseconds) that is used in RTO calculations.
Max Init Retransmits (Client)	Maximum number of attempts to establish a path connecting to a peer.
RTO Min (Server & Client)	Minimum value (in milliseconds) used for the RTO. If the computed value of RTO is less than RTO Min, the computed value is rounded up to this value.
SCTP_MAXSEG (Server & Client)	Maximum size (in bytes) of the data chunks that the SCTP message can be divided into for all the paths in an association.
RTO Max (Server & Client)	Maximum value (in milliseconds) used for RTO. If the computed value of RTO is greater than RTO Max, the computed value is rounded down to this value.
Sack_Timeout (Server & Client)	Time (in milliseconds) that the peer waits for a selective acknowledgement (SACK).

3.7 Licensing the FEPs

Prior to a FEP being able to process traffic, each IP address that is associated with a configured FEP must have a valid license. You can either have a separate license per IP address or you can have a network of licenses for multiple IP addresses. To support a network of licenses, a NetMask is defined in the license key, as shown in the following example:



□ CPF-COMMERCIAL-Traffix Systems-Demo-0-10000000-2024-01-01-1-192.168.190.64-NetMask-24-8d7e00b65ef2bcd3700781657fdc3cfd

The NetMask license mechanism supports IPV4 and IPV6.

License keys are generated and provided to you by F5 Support.

3.7.1 Adding a New License Key

Each new license key needs to be added.

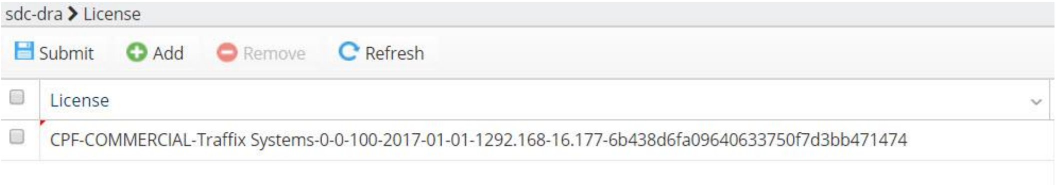
To enter a new license key:

1. Go to **Administration** > **Specific Site Settings** > **License**. The License screen is displayed:
2. Click **Add**.
3. Enter the license key provided to you by F5 Support.
4. Click **Submit**.

3.7.1.1 The License Key's Structure

The provided key represents different properties, separated by a hyphen, as shown in *Figure 26*.

Figure 26: License Key



sdc-dra > License

☐	License
☐	CPF-COMMERCIAL-Traffix Systems-0-0-100-2017-01-01-1292.168-16.177-6b438d6fa09640633750f7d3bb471474

Table 25 describes the license key properties:

Table 25: License Key Properties

License Key Property Value Example	Description
CPF	The component's name
COMMERCIAL	Evaluation/Commercial version indication
F5 Systems	The customer's name
0-0-100	The TPS (transactions per second)
2015-01-01	The license key expiration date

License Key Property Value Example	Description
192.168.16.177	IP Address
NetMask-24	Identifies the license as a network license for multiple IP addresses per FEP, either for IPV4 (value range 16-32) or for IPV6 (value range 64-128)
MD5 number	Hash used for encryption for NetMask

3.7.2 Removing a License Key

You can also remove a license key from the **License** list.

To remove a license key from the list:

1. Select the row of the license key you want to remove.
2. Click **Remove**.

   A confirmation message appears.
3. Click **OK**.

4. Configuring the SDC Flow Management

This chapter describes the message flow and how you can configure, manage, and transform messages throughout the SDC pipeline.

When SDC receives a request from a Client Peer, the request is examined, routed to its destination and transformed into the right format according to its content.

In the **Routing** tab you may define the logical sequence of conditions and actions according to which SDC routes and transforms requests and answers.

SDC's internal flow is illustrated in *Figure 27* and detailed in *Table 26*.

Figure 27: SDC Internal Flow Logic

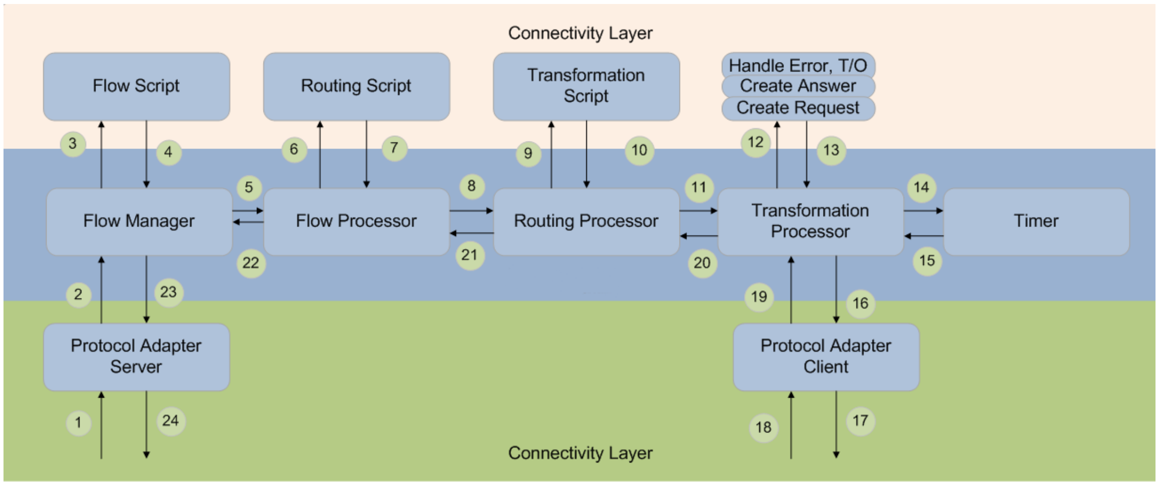


Table 26: SDC Flow Logic Legend

Event num.	Description
1	A Diameter request is received.
2 – 16	SDC interacts with user defined Business Logic to perform the preconfigured transformation to the target protocol format.
17	The transformed request is sent to the destination Server Peer.
18 or 15	A successful receipt of an answer (18) or timeout (15) takes place.

Event num.	Description
19 – 23	After successfully receiving an answer, a sequence of transformation is performed (19)-(23) to prepare and send the answer to the source where the request originated.
24	A Diameter answer is sent.

4.1 Dictionary

The **Data Dictionary** defines the format of a protocol’s messages and their validation parameters: structure, number of fields, data format, etc. Each protocol is defined with a data dictionary.

To add a data dictionary:

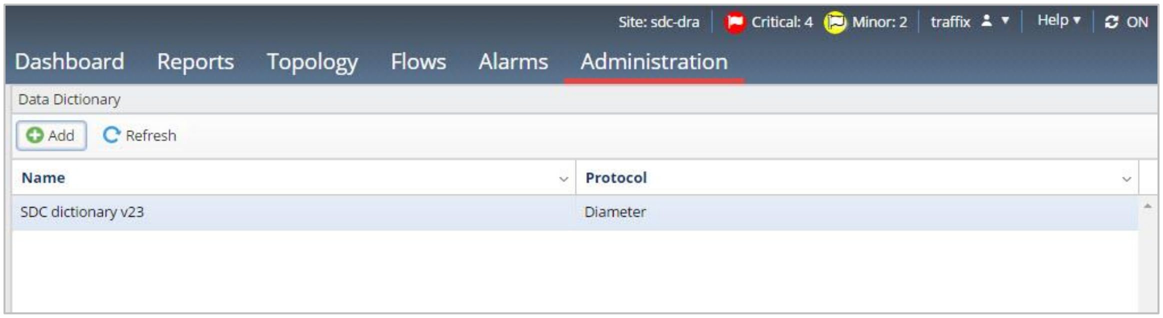
1. Go to **Administration** > **Data Dictionary**. The Data Dictionary screen displays the currently installed data dictionaries per protocol.

Note: Adding a data dictionary for a protocol that already has a data dictionary installed for it will replace the existing data dictionary with the added data dictionary.

Note: When selecting a data dictionary for RADIUS, the header message Vendor Specific Attribute now supports configurable format types. The default format number is 1. The format can now be configured between 1-4, and can be added by setting the dictionary with the vendor name with the additional number (between 1 and 4):

VENDOR <vendor id> <vendor name> [format]

Figure 28: Data Dictionary



Name	Protocol
SDC dictionary v23	Diameter

2. Click **Add**.

   The **Add Dictionary** screen appears.

3. In the **Location** field, enter the path to the data dictionary file, or click **Browse** to select the data dictionary file's location.

   Note: You can only upload a CSV file.

4. In the **Protocol** field, select the data dictionary's supported protocol.

5. Click **Save**.

   The data dictionary file now appears in the Data Dictionary list.

6. Click **Submit**.

   The data dictionary is now installed.

4.2 External Lookup Management

External lookup items allow you to run scripts to extract data from external sources such as LDAP or Coherence. You may define scripts to run upon SDC startup and shutdown which will obtain information that can be used by SDC. You may use external lookup scripts in Session Binding, for example.

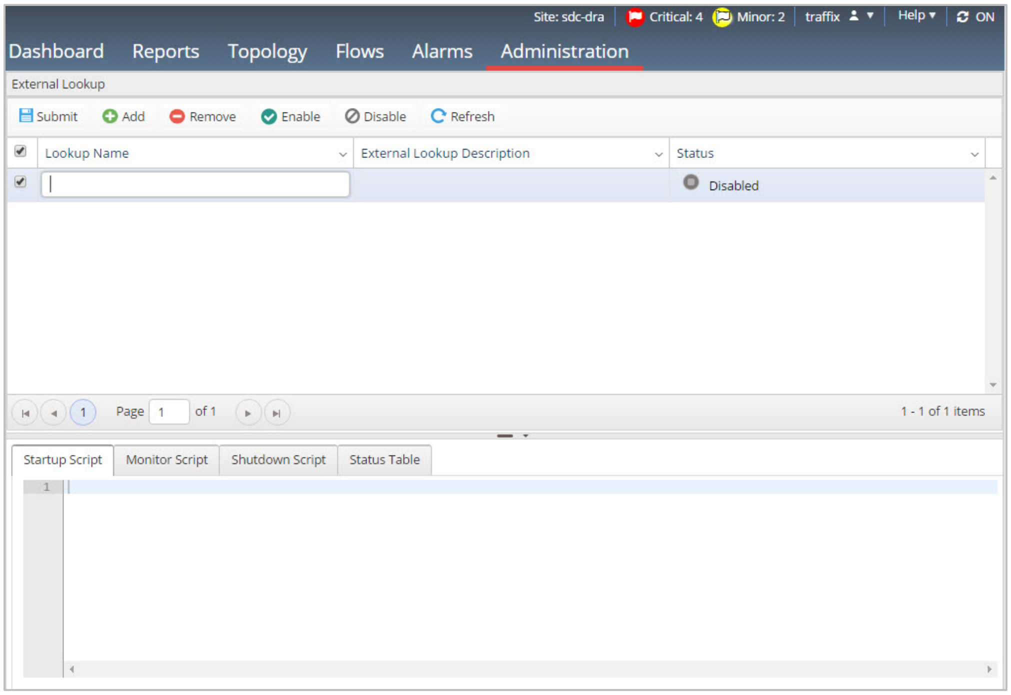
To add an external lookup item:

Go to **Administration** > **External Lookup Management**. The External Lookup Management screen is displayed.

1. Click **Add**. The Add External Lookup dialog box appears.

Figure 29: Add External Lookup

2.



3.

4. In **Lookup Name**, type in the name of the external lookup item (e.g. “LDAP”).

5. In **External Lookup Description**, enter a short text to describe the new lookup item (e.g. “Connects to LDAP and extracts IMSI”).

In **Startup Script**, set the script to run each time SDC is initiated.

The following is an example of a startup script.

```
//startup script
userTraceLogger.info("Coherence IMDB cache connection: starting.....");
def subscriberCache = com.tangosol.net.CacheFactory.getCache("SubscriberToZone");
if (subscriberCache.isActive()) {
    userTraceLogger.info("Coherence IMDB SubscriberToZone Cache connected");
```

```
}else{
    com.tangosol.net.CacheFactory.releaseCache("SubscriberToZone");
    subscriberCache = com.tangosol.net.CacheFactory.getCache("SubscriberToZone");
}
def npanxxCache = com.tangosol.net.CacheFactory.getCache("NPANXXToZone");
if (npanxxCache.isActive()) {
    userTraceLogger.info("Coherence IMDB NPANXXToZone Cache connected");
}else{
    com.tangosol.net.CacheFactory.releaseCache("NPANXXToZone");
    npanxxCache = com.tangosol.net.CacheFactory.getCache("NPANXXToZone");
}
def marketCache = com.tangosol.net.CacheFactory.getCache("MarketToZone");
if (marketCache.isActive()) {
    userTraceLogger.info("Coherence IMDB MarketToZone Cache connected");
}else{
    com.tangosol.net.CacheFactory.releaseCache("MarketToZone");
    marketCache = com.tangosol.net.CacheFactory.getCache("MarketToZone");
```

6. In **Monitor Script**, set the script to run and monitor the script’s connection with the external source and the monitoring scripts’ run interval (in Millis), as shown in the following example.

```
//monitoring script
def subscriberCache = com.tangosol.net.CacheFactory.getCache("SubscriberToZone");
if (!subscriberCache.isActive()) {
     userTraceLogger.info("Coherence IMDB SubscriberToZone Cache not accessable, re-
initiating..");
    com.tangosol.net.CacheFactory.releaseCache("SubscriberToZone");
    subscriberCache = com.tangosol.net.CacheFactory.getCache("SubscriberToZone");
return false;
}
def npanxxCache = com.tangosol.net.CacheFactory.getCache("NPANXXToZone");
if (!npanxxCache.isActive()) {
    userTraceLogger.info("Coherence IMDB NPANXXToZone Cache not accessable, re-
initiating..");
    com.tangosol.net.CacheFactory.releaseCache("NPANXXToZone");
    npanxxCache = com.tangosol.net.CacheFactory.getCache("NPANXXToZone");
return false;
}
def marketCache = com.tangosol.net.CacheFactory.getCache("MarketToZone");
if (!marketCache.isActive()) {
    userTraceLogger.info("Coherence IMDB MarketToZone Cache not accessable, re-
initiating..");
    com.tangosol.net.CacheFactory.releaseCache("MarketToZone");
    marketCache = com.tangosol.net.CacheFactory.getCache("MarketToZone");
return false;
}
return true;
```

In **Shutdown Script**, set the scripts to run each time SDC shuts down, as shown in the following example.

7.

```
//shutdown script
userTraceLogger.info("Coherence IMDB Cache releasing: started.....");
com.tangosol.net.CacheFactory.releaseCache("SubscriberToZone");
userTraceLogger.info("Coherence IMDB SubscriberToZone cache released");
com.tangosol.net.CacheFactory.releaseCache("NPANXXToZone");
userTraceLogger.info("Coherence IMDB NPANXXToZone cache released");
com.tangosol.net.CacheFactory.releaseCache("MarketToZone");
userTraceLogger.info("Coherence IMDB MarketToZone cache released");
```

Table 27 details the External LookupScript parameters.

Table 27: Lookup Script Parameters

Parameter	Type
Stack	Stack
externalLookupProperties	PropertiesOwner
UserTraceLoggerWrapper	userTraceLogger
metaData	MetaData

8.

Note: You may only call API methods associated with the parameters listed in *Table 27* (to view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

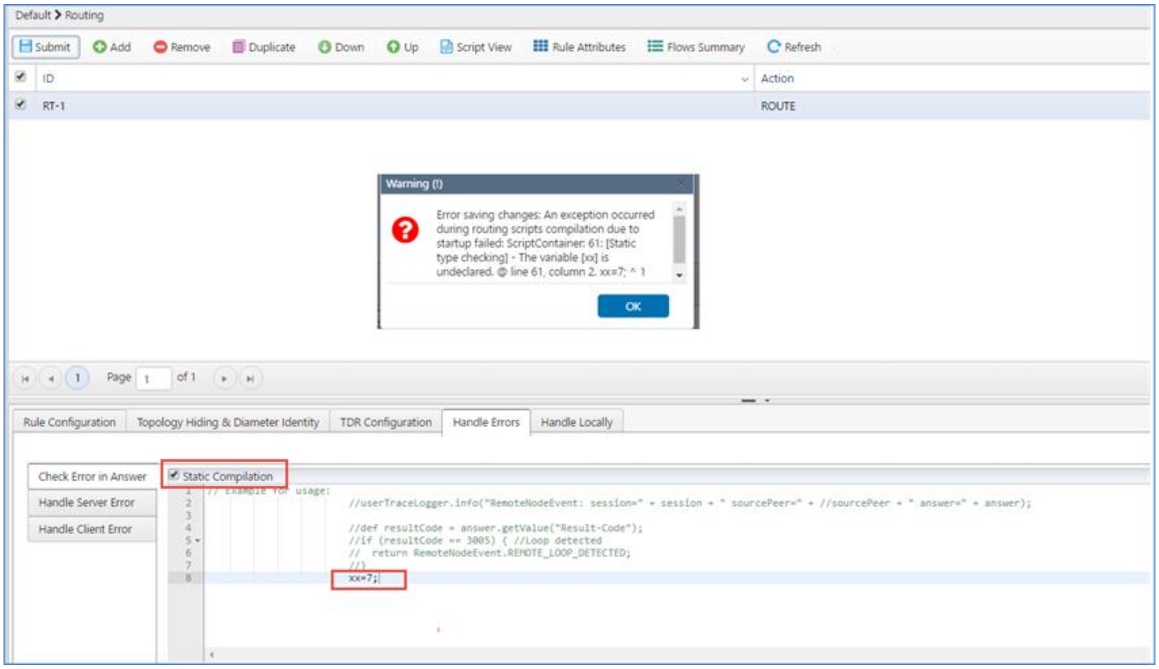
Click **Submit**. The new External Lookup item is added. You may click the item's line to edit it.

4.3 Using Scripts

To set the script actions for the routing and transformation actions, you should be acquainted with Groovy scripting language (for more information on Groovy scripting, see *http://groovy.codehaus.org/*) and the Connectivity API (to view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

Scripts applied to a Flow action (Session Management, Routing Rules, Transformation) include a static compilation option. By selecting the **Static Compilation** checkbox, you invoke a validation mechanism to check that the script content does not include any invalid parameters or syntax. If the script includes an invalid parameter, upon submitting the script, an error warning message is displayed. In addition, the validation mechanism checks that the script does not contain any API violation and all API methods are valid.

Figure 30: Static Compilation Example



Warning (!)

Error saving changes: An exception occurred during routing scripts compilation due to startup failed: ScriptContainer: 61: [Static type checking] - The variable [xx] is undeclared. @ line 61, column 2. xx=7; ^ 1

☑ Static Compilation

```
// Example for usage:
        //userTraceLogger.info("RemoteNodeEvent: session=" + session + " sourcePeer=" + //sourcePeer + " answer=" + answer);

        //def resultCode = answer.getValue("Result-Code");
        //if (resultCode == 3005) { //Loop detected
        //  return RemoteNodeEvent.REMOTE_LOOP_DETECTED;
        //}
        xx=7;
```

Statically compiled scripts use less system resources and, as a result, are executed efficiently.

4.3.1 Disabling External Lookup

By default, External Lookup is enabled. There is also the option to disable access to a specific External Lookup database.

To disable access to an External Lookup database:

1. In **Administration** > **External Lookup Management**, select a row in the table.
2. Click **Disable**.

   A confirmation message appears.
3. Click **OK**.

4.3.2 Removing an External Lookup Data Source

You can remove one of the External Lookup data sources.

To remove an External Lookup data source from the list:

1. Select the row of the External Lookup data source that you want to remove.
2. Click **Remove**.

   A confirmation message appears.
3. Click **OK**.

4.4 Flow Management

The flow management decision table is the first in a series of decision tables that incoming messages will be processed according to. Each flow includes a session management, routing rules, and transformation decision table. The flow management decision table defines which flow of session management, routing rules, and transformation decision tables an incoming message will be sent to.

Note: The Flows table cannot be edited during an upgrade of multiple SDC sites managed by an EMS site.

4.4.1 Creating a New Flow

The flow management table contains a default flow. If no rules are configured, all incoming messages are processed by the session management, routing rules, and transformation decision tables in the default flow.

To create a new flow:

1. Go to **Flows** > **Flows**.
2. Click **Add Flow**. The “Add Flow” wizard appears.
3. In the **Flow Name** field, enter the name of the flow you want to add.
4. Click **OK**.

The added flow now appears in the navigation pane, containing an empty session management, routing rules, and transformation, decision table.

4.4.2 Assigning Messages to a Flow

The flow management decision table defines the rules that will be used to decide which of the configured flows will process each incoming message.

To assign a flow to a specific message:

1. Go to **Flows** > **Flows**.
2. Configure rule attributes for the Flow rules by following the instructions in *Adding Rule Attributes*.
3. Define the rule criteria for the Flow rules by following the instructions in *Defining the Rule Criteria*.
4. From the drop-down list under **Flow Name**, select the flow that messages matching the rule criteria should be assigned to.
5. Click **Submit**.

All incoming messages that match the defined rule criteria for a specific flow will be processed according to the decision tables associated with that flow.

4.5 Session Management

The session management decision table defines how decisions will be taken for incoming messages.

You can apply a session management method to each type of session and compose special scripts that will run upon each session type creation, session update and session release. These scripts may be used to log specific transactions according to message content, for example.

You may also create rule-based session management rules. The rules consist of parameters that are defined by the Session Attributes – a list of AVP's. Each AVP is assigned a type (Boolean, regular expression, etc.).

The session management functionality defines the dependency between different sessions initiated from different remote peers which share common attributes. Bound sessions are handled as a session bundle composed of several sub-sessions.

Bound sessions are related to as Slave Sessions subject to their Master Sessions. The Master Session is the session for which the routing selection is performed based on the routing rules. Slave Sessions are applied with routing rules inherited from the Master Session.

Session management is done using Session Keys. Session Keys are sets of values extracted from different attributes (e.g. AVPs or XML attributes) of the Master Session and used to bind several session identities.

Session Lookup provides a way to customize a message's session properties, such as the routing destination. For instance, you can configure a different routing destination (i.e. OCS) for a Gy message even though its associated session, a Gx message, has a different destination (i.e. PCRF). Unlike the slave-master session relationship, after the selected session properties, such as the destination property, are copied, there is no relationship between the new session (no masterSessionId property) and the session from which only the selected properties were copied.

Note: In an SDC deployment without a central EMS configuration, all the session management configuration must be configured identically (including the same Routing and Session Management rows) in both SDC mated sites to ensure session management and binding consistency.

From an SDC Web UI, you can view the session binding rules that were configured globally from an EMS Web UI.

4.5.1 Assigning Messages to a Session Action

The following section describes how to configure the Session Management decision table to assign the desired session action rule (and apply its associated configurations) to the correct incoming messages.

Warning: When more than one flow is defined, there is more than one Session Management decision table. Verify that you are working with the Session Management decision table under the desired flow.

To assign a session management action to a specific message:

1. Go to **Flows >Flows**> **<Flow Name>** > **Session Management**. The Session Management screen is displayed.
2. Configure rule attributes for the Session Management rules by following the instructions in *Adding Rule Attributes.*
3. Define the rule criteria for the Session Management rules by following the instructions in *Defining the Rule Criteria*.
4. From the drop-down list under **Action**, select the action that messages matching the rule criteria should be assigned to.

Table 28 describes the different session actions and their configurations:

Note: If no session record (existing session, Cache, External, etc.) is found, the request will be processed by the routing decision table.

Table 28: Session Actions

Session Binding Rule	**Description**	**Method Configuration**
Cache	Indicates that the Routing is performed based on the routing rule and the routing decision creates a binding record entry holding the relevant keys. You must specify the key sets (zero or more) that can be used for resolving this binding record.	Session Keys Session Properties Session Life Cycle Scripts
External	Indicates that the routing decision of this session creates a binding record holding the relevant keys. The destination is selected by performing a lookup in an external data source., You must specify the properties, script and the key sets (zero or more) that can be used for resolving this binding record.	External Lookup Session Keys Session Properties Session Life Cycle Scripts
Resolve	Indicates that cached routing decisions should be used for this session. You must specify the key set that will be used for resolving the binding record. Note: When executing a transactionEvent.setStateless(true) script (Routing>Transaction>Pre-Routing) on a resolve (or slave) session, it is considered stateless. For each transaction, the system	Binding Key Selection Session Properties Session Life Cycle Scripts

Session Binding Rule	Description	Method Configuration
	will always check Session Repository for its master's state (based on its binding key) and never for the state of the session (based on its session ID).	
Resolve or External	Indicates a combination of the External and Cache options. If possible, the destination is selected by performing a lookup in external data source. Else, Cached routing decision is used.	External Lookup Binding Key Selection Session Properties Session Life Cycle Scripts
Resolve or Cache	Indicates a combination of the Resolve and Cache options. If possible, Cached routing decision is used. Else, Routing is performed based on the routing rules.	Binding Key Selection Session Properties Session Life Cycle Scripts
Session Lookup	Indicates that the routing session for this session is based on selected properties that were copied from a cached session.	Lookup Properties Session Keys Session Properties Session Life Cycle Scripts
No Binding	Indicates no binding. In this case only the Life-Cycle scripts are applied to the matching session. This is the default action.	Session Keys Session Properties Session Life Cycle Scripts

4.5.1.1 Defining Session Keys

A session record includes the session keys related to a selected session. A session key consists of a name and its content. The keys are used to lookup the session data and session destination for ongoing transactions within a session, as well as, a lookup of a master session when a slave session arrives. The session keys are saved in the Session Repository. You only define the binding keys when you select a **Cache** (Master), **External**, Session

Lookup rule. The session ID is always the first session key and you can add up to four other keys, such as IPV6 or an IMSI.

To add a session key:

1. Click **Binding Record Definition**.
2. Click **Add** to create a new key saved to the session's cache.
3. Enter the **Key Name** and its **Content**.

4.5.1.2 Selecting a Defined Binding Key

When configuring a **Resolve** binding rule, you need to select one of the session keys that was defined for a related Cache or External session rule.

To select a defined binding key:

1. Under **Binding Key Selection:**
   a. In **Defined Keys**, select from the drop-down list the key against which you want the resolved session to be compared.
   b. In **Key Content**, the selected binding key content is displayed.

Note: Sessions which share the same key value as the master session will bound to it. If not, a new Cache binding rule will be executed, recording the entered values: key name and content.

4.5.1.3 Configuring a Persistency Policy

The persistence policy determines if session data will be saved in the Session Repository. Session data consists of all the relevant information about the session such as the session ID, Origin Host, Binding Keys, and Session Destination. For Cache Binding Rules, session data must be persisted as by definition, Master sessions set that all messages within the session are routed to the same destination and a **Don't Persist** means that different messages within the same session are routed to different destinations. When configuring a persistency for a Geo-redundant deployment, the session should be configured as **Persist and Replicate**.

To configure a Persistence Policy:

1. Go to **Flows > Flows > <Flow Name> > Session Management** and select a session action rule row.
2. Select **Session Properties**.
3. Under **Session Persistence Policy**, select one of the following options:
   - **Don't Persist -** to not save the session data
   - **Persist** - to save the session data in a Session Repository in a single site SDC deployment
   - **Persist and Replicate -** to save the session data in a Session repository and replicate it to another Session repository instance on an SDC mated site

Note: If you select **Don't Persist**, then each time a message of the same session is routed, it is to a different destination.

If you want the session to be replicated to a mated SDC site as part of the Session Repository Site Replication feature, you must select **Persist and Replicate**.

To have a slave session be routed to the mated site, even when the master session is terminated, configure the isLocalRoute, in the traffix_cpf_init script, to false. A default value of true means that the slave session will only be routed according to the local site's routing rules.

Do not **Persist** HTTP sessions.

4.5.1.4 Configuring End of Session Policy

You can configure the end of session policy for sessions that are persisted to the Session Repository, by setting the amount of time that a session is timed out, what triggers the time out period and what happens to a session when it is terminated.

Note: If a session is configured as **Don't Persist** then the session timeout and end of session policy configurations are grayed out as they are not relevant.

To configure the end of session policy:

Go to **Flows > Flows > <Flow Name> > Session Management** and select a session action rule row.

1. Select **Session Properties > Persist/ Persist and Replicate**.

2.

3. In the **Time Out** drop-down, select the time and time frame (in the predefined time units) after which the session is released. Requests of the same session are routed to the same destination as the destination of the first request within the session. If a session has timed-out, the requests' destination is reselected according to SDC's rules.

4. Select **Delete Session upon Termination Event** to release Diameter or RADIUS sessions upon a termination message (CCA (272) , requestType = TERMINATION_REQUEST - 3 or EVENT_REQUEST- 4 ACA(271), requestType = EVENT RECORD -1 or STOP_RECORD - 4) retrieval (rather than upon timeout).

Note: The default is that **Delete Session upon Termination Event** is selected.

Selecting the **Delete Session upon Termination Event** checkbox is not relevant for a REJECT Routing Action or when a Handle Locally script is configured (FORWARD, ROUTE, SITE PROXY Routing Actions).

5. Sessions configured to **Delete Session upon Termination Event**, can be released and deleted, even when there is no Post-Routing Transformation.

Select **Reset Session Timeout upon "Get Session Data" event** to reset a timeout upon every session data withdrawal in addition to resetting the session timeout regularly upon session data update.

Note: The default is that **Reset Session Timeout upon "Get Session Data" event** is selected.

4.5.1.5 Configuring Session Destination Decision Policy

Events within the same session, by default, are routed to the same destination peer. However, this is configurable and you can configure a session rule to make new routing destination decisions for each session event.

1. Go to **Flows > Flows > <Flow Name> > Session Management** and select a session action rule row.
2. Select **Session Properties > Persist/ Persist and Replicate**.
3. Select **Decide Routing on each Session Event** to have a new routing decision be made for each event in the session.

Note: The default is that **Decide Routing on each Session Event** is selected and grayed out when the **Don't Persist** Persistence Policy is selected. This means that for non-persisted sessions, the default is that, the destination peer information is not saved and applied to other events in the session. The default is that **Decide Routing on each Session Event** is not selected when the **Persist/ Persist and Replicate** Persistence Policy is selected.

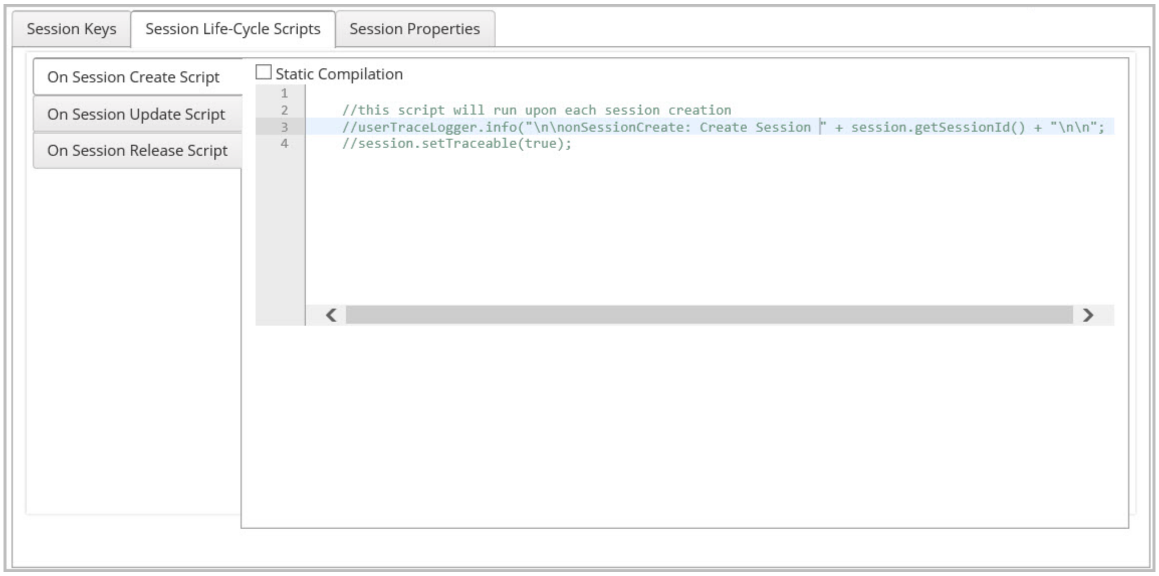
4.5.1.6 Configuring Session Life-Cycle Scripts

Life-Cycle scripts run upon session creation, session update, and session termination. These scripts may be used to log specific transactions according to message content, for example.

To implement the On Session Create script:

1. Go to **Session Life-Cycle Scripts** > **Session Create Script**.

Figure 31: On Session Create Script



□ Static Compilation

```

    //this script will run upon each session creation
    //userTraceLogger.info("\n\nonSessionCreate: Create Session " + session.getSessionId() + "\n\n";
    //session.setTraceable(true);
```

2. Set the script to run each time a new session is created.

Table 29 details the parameters SDC provides to the scripts.

Table 29: On Session Create Script Parameters

Parameter	Type
Session	Session
Message	Message
userTraceLogger	UserTraceLoggerWrapper
metadata	MetaData

Note: You may only call API methods associated with the parameters in Table 29. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

The following is an example of an On Session Create Script:

```
def vasId = message.get("ServiceInformation").get("MMSInformation").get("VAS-ID").get();
```

```
if (vasId.equals("MMS")) {
session.setTraceable(true);
}
```

To implement the On Session Update script:

Click **On Session Update Script**.

Set the script to run each time a new session is updated.

1.

Table 30 details the parameters used in the On Session Update script.

2.

Table 30: On Session Update Script Parameters

Parameter	Type
session	Session
message	Message
Stack	Stack
userTraceLogger	UserTraceLoggerWrapper
metaData	MetaData

Note: You may only call API methods associated with the parameters in *Table 30*. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

The following is an example of an On Session Update Script:

```
<OnSessionUpdate>
   <![CDATA[
   def provider = StorageProviderFactory.getInstance();
  def routingTable = provider.getUserTable("RoutingTable");
  def sessionId = session.getSessionId();
  def key = session.getContextId();
  def tmpPeer = session.getDestinationPeer();
  userTraceLogger.log(Level.WARN, "Extracted peer: " + tmpPeer.getName());
  def newDestinationPeer= new String(tmpPeer.getName());
  def list = routingTable.get(key);
  if (list !=null) {
    if(!list[0].equals(newDestinationPeer)){
```

```
    userTraceLogger.log(Level.WARN, "Peer per session: " + sessionId + " was changed
and requires update for a key: " + list);
    list[0] = newDestinationPeer;
       for (def i = 2; i < list.size(); i++) {
       userTraceLogger.log(Level.WARN, "Changing destination peer per session:" +
list[i] + "to a new peer " + newDestinationPeer);
       if (!sessionId.equals(list[i])) {
          def extractedSession = stack.getStorage().getSession(list[i]);
          extractedSession.setDestinationPeer(tmpPeer);
       }
        }
      } else
     userTraceLogger.log(Level.WARN, " No action -> Destination peer wasn't
changed");
    } else {
  userTraceLogger.log(Level.WARN, " no key was found for the session:" +sessionId);
  }
  return null;
   ]]>
  </OnSessionUpdate>
```

To implement the On Session Release script:

1. Note: This script is called upon calling session release and session timeout.

2. Click **On Session Release Script**.

   Set the script to run each time a session is released.

Table 31 details the On Session Release Script Parameters.

Table 31: On Session Release Script Parameters

Parameter	**Type**
Session	Session
userTraceLogger	UserTraceLoggerWrapper
metaData	MetaData

Note: You may only call API methods associated with the parameters in *Table 31*. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

The following is an example of an On Session Release Script:

```
def vasId = message.get("ServiceInformation").get("MMSInformation").get("VAS-
ID").get();

if (vasId.equals("MMS")) {
userTraceLogger.trace("Done with session " + session.getSessionId());
}:
```

4.5.1.7 Configuring A Session Lookup Routing Rule

Selecting a **Session Lookup** action allows you to customize a session property (i.e. destination) for unique routing scenarios for a message that you want sometimes, in certain scenarios, to be routed not according to a master session's session property, (i.e. destination). This is done, by defining a session lookup rule with defined keys that the SDC uses to check and find a matching session in the session repository from which it then copies the relevant session properties with a defined customized value. That defined customized value is how the session knows to be routed differently than the session from which it copied the session property. Then, when routing you can add a routing rule with a relevant rule attribute and/or script that is associated with the customized session property so that a message that meets the defined session management and routing rules criteria will be routed according to the customized session property.

A session configured with a Session Lookup action can be persisted or not. Session Lookup
1. is enabled for both server and client side requests.

To configure a Session Lookup binding rule for a specific message:

Go to **Flows** > **Flows** > **Session Management** and select a session binding rule in the table and then from the drop-down list under **Action**, select **Session Lookup**.

Under **Lookup Properties**, in **Defined Keys** and **Key Content**, enter the name of the session key, (i.e. IMSI) and its value, respectively, that you want to use to search for a matching master session.

2. From the matched session, the selected session properties are copied.

Under **Session Properties Mapping**, click **Add.**

3. Under **Session Property**, enter the name that you want to call the customized session property.

4. Under **Session Property – From Session**, enter the session property name from the master session that you want to copy for it to be applied during routing.

5. For example, sessionData.OCS_Name

6. Continue with the other Session Management rule configurations: **Session Properties, Session Keys**, and **Session Life-Cycle Scripts** for the session management rule.

Note: In **Session Life-Cycle Scripts**, you can customize conditions for the session. For session lookup, the On Session Update Script would be relevant for Gy update messages.

7. Click **Submit**.

To apply the customized session property to a routing rule:

1. Go to **Flows** > **Flows** > **Routing Rules** and click **Add**.

2. Create a Rule Attribute with the customized session property and define its value for how you want the message routed. For more information on configuring rule attributes, see *Adding Rule Attributes*.

4.5.1.8 Configuring an External Session Management Routing Rule

When configuring session management rules for how decisions will be taken for incoming messages, you can apply an **External** session binding action. Selecting an **External** session binding action means that the routing destination is selected by performing a lookup in an external data source, as defined in an External Lookup Script.

In addition to defining which external database repository will be used, you need to configure the Session Properties, Session Keys, and Session Life-Cycle Scripts for the session management rule.

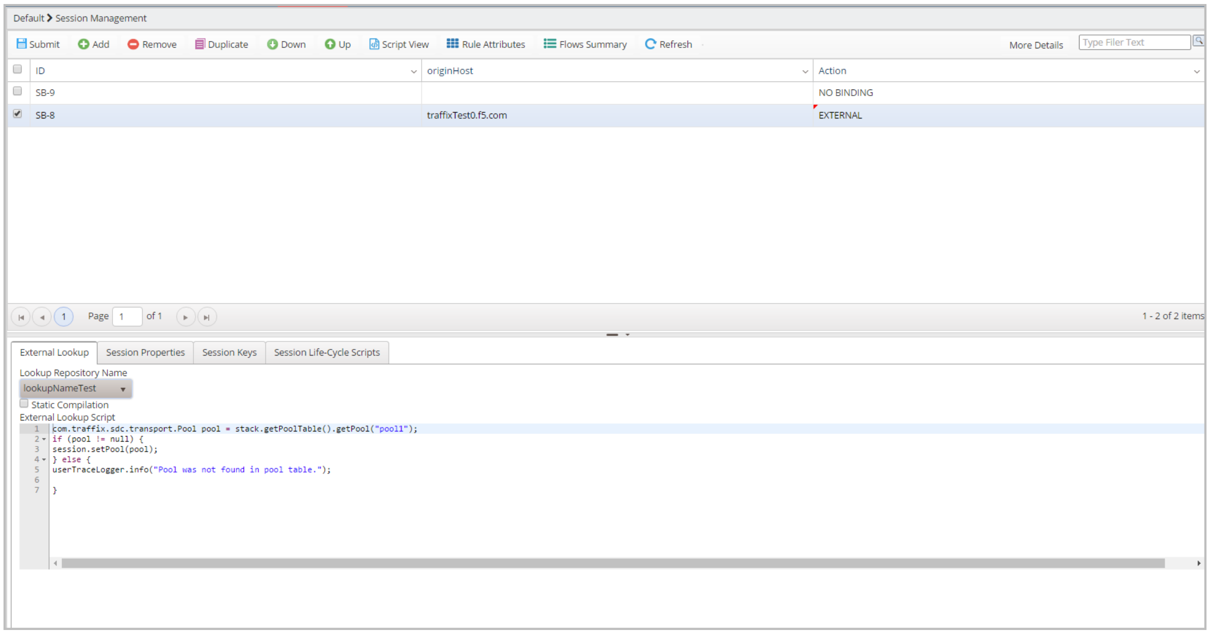
4.5.1.8.1 Configuring an External Lookup Script Session Management Rule

The script defines the way to handle the session.

To assign an External Lookup Script session binding rule:

1. From the drop-down list under **Action**, select **External**.
2. In the **External Lookup** tab, under **Lookup Repository Name** drop-down select **External Lookup Script**.

Figure 32: External Lookup Script



	ID	originHost	Action
□	SB-9		NO BINDING
☑	SB-8	traffixTest0.f5.com	EXTERNAL

Lookup Repository Name: lookupNameTest

□ Static Compilation

External Lookup Script

```
com.traffix.sdc.transport.Pool pool = stack.getPoolTable().getPool("pool1");
if (pool != null) {
session.setPool(pool);
} else {
userTraceLogger.info("Pool was not found in pool table.");

}
```

3. In **Lookup Repository Name**, select an external repository from the drop-down list. For information on external lookup scripts, see *External Lookup Management*.

Type in the script.

Table 32 details the parameters SDC provides to the scripts:

4.

Table 32: External Lookup Script Parameters

Parameter	**Type**
session	Session
message	Message
UserTraceLoggerWrapper	userTraceLogger
metaData	MetaData
externalLookupProperties	PropertiesOwner

Note: You may only call API methods associated with the parameters in *Table 32*. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

The following is an example of an External Lookup Script:

```
//Session Binding, External Lookup Script:
userTraceLogger.info("\n\nSessionBinding External Lookup looking for imsi\n\n")
    def initialContext = externalLookupProperties.getProperty("initialContext");  //
ADDED - Retrieve the connection
userTraceLogger.info("\n\nRetreived connection"+initialContext+"\n\n")
    String SERVER_POOL = "serverPool";
    String dn = "ou=subscribers,dc=oft,dc=4g,dc=orange,dc=com";

def subscriptionId = message.get("Subscription-Id");
// initialize the session values
session.setProperty("imsi",-1);
def subscriptionIdData;
while (subscriptionId != null) {
               def subscriptionIdType = (Integer)
subscriptionId.getValue("Subscription-Id-Type");
               subscriptionIdData =
Long.valueOf(subscriptionId.getValue("Subscription-Id-Data"));

               // Subscription-Id-Data does not contain value
```

```
                if (subscriptionIdData == null) {
                                subscriptionId = subscriptionId.next();
                                continue;
                }
                // Subscription-Id-Type contains 1 (IMSI)
                if (subscriptionIdType == 1) {
                                session.setProperty("imsi", subscriptionIdData);
                }

                subscriptionId = subscriptionId.next();
}
userTraceLogger.info("extracted imsi is: " + subscriptionIdData);
String imsi = subscriptionIdData;

      javax.naming.directory.SearchControls ctls = new
javax.naming.directory.SearchControls();
      String[] arr = new String[1];
      arr[0] = SERVER_POOL;
      ctls.setReturningAttributes(arr);
      ctls.setSearchScope(javax.naming.directory.SearchControls.SUBTREE_SCOPE);

      String filter = "imsi={0}";
      Object[] imsiArr = new Object[1];
      imsiArr[0] = imsi;
userTraceLogger.info("before querying... ");
      javax.naming.NamingEnumeration<javax.naming.directory.SearchResult>
enumeration = initialContext.search(dn, filter, imsiArr, ctls);
userTraceLogger.info("...after querying. enumeration=" + enumeration);
      if (enumeration == null || !enumeration.hasMoreElements()) {
        userTraceLogger.info("no pool was found for imsi " + imsi);
        return;
      }
      javax.naming.directory.SearchResult searchResult = enumeration.next();
      javax.naming.directory.Attributes attributes = searchResult.getAttributes();
      javax.naming.directory.Attribute attribute = attributes.get(SERVER_POOL);
      String shapingTemplate = (String) attribute.get();
      userTraceLogger.info("retrieved pool name from ldap lib is " +
shapingTemplate);
// ADDED 2  - Setting session's pool name
                 com.traffix.openblox.core.transport.Pool pool =
session.flowManager.getPoolTable().getPool(shapingTemplate);
                 if (pool != null) {
```

```
                                session.setPool(pool);
                } else {
                                userTraceLogger.info("Pool "+ shapingTemplate +" was
not found in pool table.");
                }
```

4.6 Routing

The routing table defines the routing process for each message received by the SDC site.

Note: While the functionalities described in this section can be configured in both SDC and EMS Web UI, it is recommended to perform these configurations globally using the EMS Web UI. When the EMS Web UI is active, it is not possible to use the SDC Web UIs.

Note: Before SDC can process traffic, you need to add a license key for each FEP VIP. For more information about licensing, see *Licensing the FEPs*.

4.6.1 Assigning Messages to a Routing Rules Action

The following section describes how to configure the Routing Rules decision table to assign the desired routing rule (and apply its associated configurations) to the correct incoming messages.

Warning: When more than one flow is defined, there is more than one Routing Rule decision table. Verify that you are working with the Routing Rule decision table under the desired flow.

1.

To assign a routing action to a specific message:

2.

Go to **Flows** > **Flows** > **<Flow Name>** > **Routing Rules**. The Routing Rules screen is displayed.

Configure rule attributes for the Routing Rules by following the instructions in *Adding Rule Attributes.*

Define the rule criteria for the Routing Rules by following the instructions in *Defining the Rule Criteria*.

3. From the drop-down list under **Action**, select the routing action that messages matching the rule criteria should be assigned to.

4. *Table 33* describes the policies and details the necessary configurations and scripts for each action. These configurations and scripts appear as tabs below the decision tab when an action is selected.

Table 33: Action Descriptions

Action	Description	Available Configuration Parameters/Scripts
Route	Routes the request to one of the specified Pools.	**Rule Configuration** **Topology Hiding/Diameter Identity** **TDR Configuration** **Check Error in Answer** **Handle Server Error** **Handle Client Error** **Handle Locally**
Discard	Silently discards the request.	**Rule Configuration**
Forward	Forwards the request to a peer or a pool (as configured in the **Rule Configuration** tab).	**Rule Configuration** **Diameter Identity** **Check Error in Answer** **Handle Server Error** **Handle Client Error** **Handle Locally**
Redirect	Sends a redirect answer with a configured server name.	**Rule Configuration** **Diameter Identity** **Redirect**

Action	Description	Available Configuration Parameters/Scripts
Reject	Performs a local termination with an error result (the result code should be configured).	**Rule Configuration** **Handle Reject**
Site Proxy	Routes the request to a remote site.	**Rule Configuration** **Check Error in Answer** **Handle Server Error** **Handle Client Error** **Handle Locally**
Terminate	Performs a local termination with a success result code (2001)	**Rule Configuration** **Create Message Locally**
Resolve & Route	Resolves and routes the request by a designated DNS server	**DNS Resolving** **Rule Configuration** **Topology Hiding/Diameter Identity** **Handle Server Error** **Handle Client Error**

4.6.1.1 Defining Rule Configuration Parameters

Each routing rule may be individually configured to determine the pools to which the message is routed, the number of resend attempts, etc.

1.

To configure the Rule Configuration parameters:

Under **Flows > Flows > <Flow Name> > Routing Rules**, select a Routing Rule and depending on which Routing Action was selected, you can configure the Routing Rule according to the parameters described in *Table 34*.

Table 34: Routing Rule Configuration Parameters

Parameter	Definition	Default Value	Note
Pools (**Route**)	Select the pool/s to which messages which match the rule's criteria are sent		In upgraded SDC sites, when adding a new flow, make sure that at least one pool is defined for this parameter.
Load Balancing Policy	Messages can be routed between multiple pools using either the **By Precedence** or the **Round Robin** policy.	**By Precedence**	The **Round Robin** policy routes messages to all pools equally.
Route Ongoing Session Messages to Selected Pool Only	Set option that ongoing session messages are forwarded to another pool when the originally selected pool is out of service	**False (checkbox not selected)**	Default setting means that the message is routed to another pool.
Max Resend Attempts (**Forward**, **Route**, **Resolve & Route**)	Set the maximum number of request sending retries, in case it fails		This parameter affects the entire Pool.
Delay Between Attempts (**Forward**, **Route**, **Resolve & Route**)	Set the time difference between one resend attempt and another	0	
Assign Notification Pool	Sets a routing rule So that a message is copied to a notification pool. When selected, the Notification Pool configuration pane opens.		For more information about configuring a routing rule to a notification pool, see *Configuring a Notification Pool*
Properties Configuration Table	Sets parameters to a routing rule: envelope-for a transaction only		

Parameter	Definition	Default Value	Note
	session-persisted in Session Repository		

4.6.1.2 Configuring Diameter Identity Routing Behavior

There is an option to define specific values to replace the values of the message’s origin-host and origin-realm AVPs. By default, the message’s origin-host AVP value is the name of the message’s virtual server, and the message’s origin-realm AVP value is configured per FEP and is taken from the FEP that the virtual server is configured to use.

You can configure the **Diameter Identity** policy for a rule’s messages, by defining if and when to replace the message’s default origin-host and origin-realm AVP values with the values configured in the peer profile, as well as persistency policies in a case of a server or SDC failover. The Destination Realm/Host data are saved in the persisted session data in the Session Repository. The session, which contains the Destination Realm/Host is replicated to the mated site, so in a case of site failover, the information remains available As a result, if you want to make sure the Destination Realm/Host data is preserved in a case of a failover, you must configure persistency for the selected rule.

Note: Configuring the Diameter Identity policy is disabled when Topology Hiding is enabled and when the routing rule is defined with either the Discard or Site Proxy actions.

To set the Diameter Identity policy:

1. Go to **Flows** > **Flows** > **<Flow Name >** > **Routing Rule**.
2. Select the relevant routing rule and then **Topology Hiding/Diameter Identity**.
3. Select **Disable** in the **Topology Hiding** pane.
4. Under **Diameter Identity**, select one of the following from the drop-down list:

▪ **Relay** – All the requests or answers will be forwarded without any modification.

▪ **Client Side Proxy** – used to abstract the server from clients.

- **Full Proxy** – used to abstract the servers from the clients and clients from the servers.
- **Roaming Proxy** – used to abstract the servers from the clients and clients from the servers in roaming use cases.

Note: Geo-redundant operators with two MMEs should configure two different peer profiles for each MME.

To configure server failover behavior:

Note: Configuring the **Server Failover** Policy is disabled when the Diameter Identity Policy is defined as **Relay**.

1. Under **Server Failover**, select **Keep Destination-Realm for session fail over** and **Keep Destination-Host for session fail over**.

   When selected, the destination-Realm/Host that is sent to the destination server will also be sent to the destination server chosen after a session failover. If not selected, the destination-Realm/Host that will be sent to each destination is the one that was learned during the capabilities exchange.

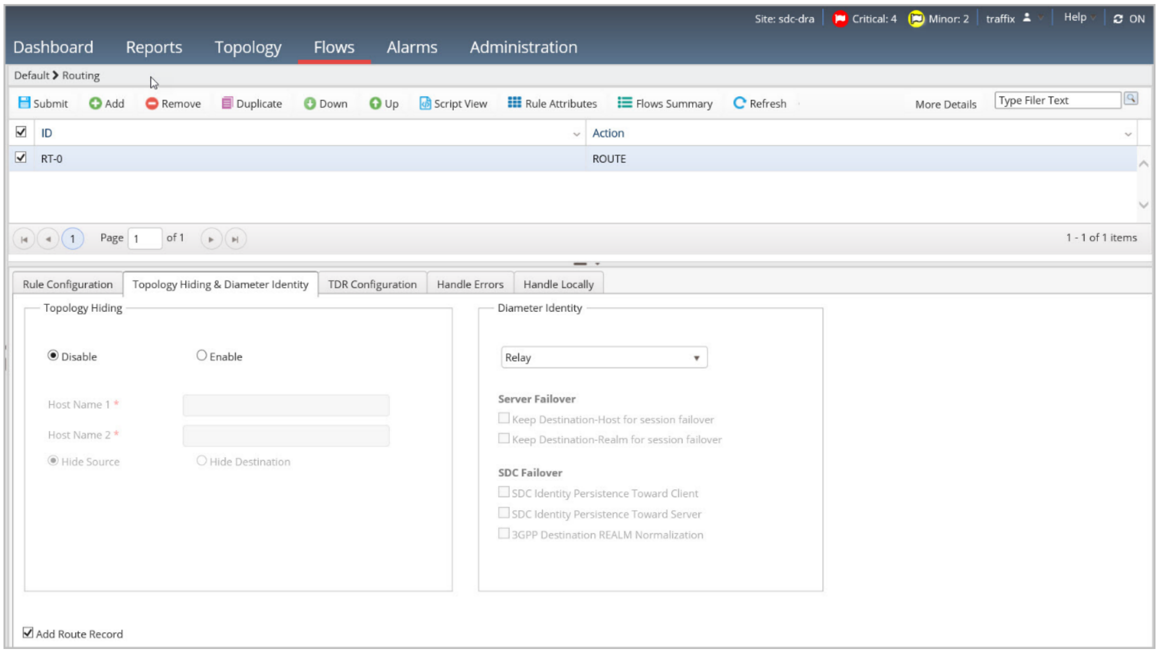
For Client Side and Full Proxy Diameter Identity policies, you can select the option to persist and replicate the Diameter Identity Policy to a replicated SDC site for an existing session in the event of a session failover scenario.

1.

To enable the persistence option in an SDC failover:

Under **SDC Failover**, select **SDC Identity Persistence Toward Client/Server**.

Figure 33: SDC Diameter Identity Persistence



☑ ID	Action
☑ RT-0	ROUTE

Topology Hiding

● Disable ○ Enable

Host Name 1 *

Host Name 2 *

● Hide Source ○ Hide Destination

Diameter Identity

Relay

Server Failover

□ Keep Destination-Host for session failover

□ Keep Destination-Realm for session failover

SDC Failover

□ SDC Identity Persistence Toward Client

□ SDC Identity Persistence Toward Server

□ 3GPP Destination REALM Normalization

☑ Add Route Record

To redefine the destination realm:

1.

Select **3GPP Destination REALM Normalization.**

When selected, the MNC and MCC is extracted from the IMSI and destination realm is changed to epc.mncXXX.mccYYY.3gppnetwork.org for every ULR message.

1.

To add a Route-Record AVP:

Select **Add Route Record**.

When selected, the SDC adds a Route-Record AVP to each received request. The Route-Record AVP contains the name of the remote peer that the request originated from, taken during the capabilities exchange, to prevent routing loops.

4.6.1.3 Configuring Topology Hiding

This feature is relevant for roaming scenarios when you want to hide either the source (MMEs) or destination (HHS) peer network topology per routing rule. When enabled, you can define the relevant transformations that hide the identity of each network element during routing. This is done by defining dummy address values for the origin host values. The routing data, post-transformation (current MME, previous MME, current dummy address in use) is saved on a subscriber level, per IMSI in the Session Repository user tables.

The following AVPs are transformed as part of the Topology Hiding:

Origin Host (set to the current dummy address), Session ID, Proxy-Host (set to the current dummy address), Error Reporting Node (set to the current dummy address), Route-Record (set to the current dummy address)

Note: Topology Hiding is disabled by default.

To enable Topology Hiding:

1. Go to **Flows** > **Flows** > **<Flow Name>** > **Routing Rule**.
2. Select the relevant routing rule and then **Topology Hiding/Diameter Identity**.
3. Select **Enable** in the Topology Hiding pane.

To hide the source or destination peer network topology name:

1. In the **Host Name 1** and **Host Name 2** fields, type a unique name that will be used as the "dummy" address.

   Note: Each new "dummy" host name must be unique. The format of the host names should be one word without any spaces or delimiters, such as a “.” If hiding the source network information (MMEs), then you need to enter two dummy addresses (**Host Name 1** and **Host Name 2**). If hiding the destination network information (HHS), then you need to only enter one dummy address.

2. Select either **Hide Source** or **Hide Destination** depending on which network information you want to hide:

- **Hide Source** – Hides the network information for incoming requests.
- **Hide Destination** – Hides the network information for incoming answers.

4.6.1.4 Defining TDRs

By default, the SDC collects and displays information for specific message AVPs. Using the **Create Transaction Data Record** table, you can add five additional AVPs for the SDC to this default setting for each defined routing rule.

To define TDR for additional AVPs:

1. Go to **Flows> Flows > <Flow Name> Routing Rules>TDR Configuration** and select **Create Transaction Data Record**. The table shows five user-defined AVPs that will be added to the information displayed in the Reports screens.
2. In the **Name** field, enter any user friendly value. This value is only used by you for reference, and will appear in the TDR reports as AVP1 through AVP5.

   Note: A Framed-IP-Address **Name** (for RADIUS requests) is not supported.
3. In the **Value** field, enter the AVP that you want to add to the default set of TDR AVPs.

   Note: You can also define TDRs to be generated based on a peer profile (**Topology>Peer Profiles>Diameter Configuration**) without the option of adding additional AVPs.

4.6.1.5 Defining Routing Scripts

The following section describes the scripts that are invoked upon action execution and the parameters provided to them by SDC.

- **Check Error in Answer**

In **Check Error in Answer**, define a rule for when an answer is sent back to the Client Peer or Server Peer (through SDC) and is indicated as an error. This option is available when selecting **Forward**, **Route**, **Site Proxy**, or **Resolve and Route** actions.

def resultCode = answer.get("Result-Code")

```
if (resultCode == 4012){
    return RemoteNodeEvent.TOO_BUSY;
}
return RemoteNodeEvent.OK;
```

Table 35 lists the possible returned values which may indicate an error in answer. You may build a suitable answer to the Client Peer, in accordance with the exact error case:

The following returned values have corresponding SNMP traps that can be triggered based on a configured rule for different result codes in the Check Error in Answer script. Refer to the *F5 SDC SNMP Guide*, for more details about each of the corresponding SNMP traps.

The following is an example of defining a returned value for a specific result code in **Check Error in Answer** script:

```
def resultCode = answer.get("Result-Code")
if (resultCode == 4012){
    return RemoteNodeEvent.TOO_BUSY;
}
return RemoteNodeEvent.OK;
```

Table 35: Check Error in Answer Returned Value

Returned Value	**Description**
RemoteNodeEvent.OK	The answer is transformed to the client.
RemoteNodeEvent.REMOTE_LOOP_DETECTED	The request is identified as a loop. The Handle

Returned Value	Description
	Server/Client Server Error script is invoked.
RemoteNodeEvent.NO_REMOTE_EVENT	All received answers are discarded and the answer is created locally in the Handle Server Error script.
RemoteNodeEvent.REDIRECT	A new Pool must be set. The request will be resent to the new Pool according to its policy.
RemoteNodeEvent.REQUEST_REJECTED	The request is rejected by the server. The request will NOT be resent according to the routing Resend parameter. Handle Server Error script is invoked.
RemoteNodeEvent.TOO_BUSY	A server error. The Request will be resent according to the routing resend parameters.
RemoteNodeEvent_APPLICATION_ERROR	Indicates an application error.
RemoteNodeEvent _SDC_APPLICATION_ERROR	The SDC could not process the request due to an SDC application-based error. For example, a CPF-based error or a FEP-based error.

Returned Value	Description
RemoteNodeEvent_SDC_OVERLOAD	The SDC could not process the request because an internal queue was full.
RemoteNodeEvent_USER_INPUT_ERROR	The SDC could not process the request, either because it did not match any routing rule, or an error occurred when invoking one of the processing scripts.
RemoteNodeEvent_MISSING_ROUTING_ DISCARDS	The SDC could not find a routing rule to match the request.
RemoteNodeEvent_PEER_RATE_LIMIT_EXCEEDED	The SDC could not process the request because the peer had reached its defined rate limit
RemoteNodeEvent_PROCESS_RATE_LIMIT_EXCEEDED	The SDC could not process the request because the SDC had reached its defined rate limit

Note: If the answer is indicated as an error, it is registered to a special error counter that eventually indicates the Server Peer’s inability to handle requests. In this case, the Remote Peer is out of service for a predefined time period.

The answer parameter affects the Remote Peer, but does not affect the entire Pool. That is, the number of errors is accumulated per Remote Peer.

Table 36 shows the **Check Error in Answer script** parameters.

Table 36: Check Error in Answer Script Parameters

Check Error in Answer Script's Returned Value Type: RemoteNodeEvent	
Parameter	**Type**
answer	Message
userTraceLogger	UserTraceLoggerWrapper
metaData	MetaData

Note: You may only call API methods associated with the parameters in *Table 36*. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

- **Handle Server Error**

In **Handle Server Error**, define a script to be invoked when the **Maximum number of Resend Attempts** has been exceeded or the Server Peer has sent an Answer indicating an error. This option is available when selecting **Forward**, **Route**, **Site Proxy**, or **Resolve and Route** actions.

Note: You may choose to act according to the specific error event that was previously detected (see **Check Error in Answer** script). This script is invoked when SDC routes an error message to a client peer, (as the destination peer).

Table 37 shows the **Handle Server Error** script parameters.

Table 37: Handle Server Error Script Parameters

Handle Server Error Script's Returned Value Type: Message	
Parameter	**Type**
Event	RemoteNodeEvent

Handle Server Error Script’s Returned Value Type: Message	
session	Session
requestFromServer	Message
requestToClient	Message
answerFromServer	Message
userTraceLogger	UserTraceLoggerWrapper
metaData	MetaData

Note: You may only call API methods associated with the parameters in *Table 37*. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

The following is an example of a **Handle Server Error** script:

```
return answerFromServer;
// or:
// if (event == RemoteNodeEvent.TOO_BUSY)
//                                                                return
requestFromClient.createAnswer(3004L);
// else
//                                                                return
requestFromClient.createAnswer(5012L);
```

- **Handle Client Error**

In Handle Client Error, define a script to perform in case the **Maximum number of Resend Attempts** has been exceeded or the Client Peer has sent an Answer indicating an error. This option is available when selecting **Forward**, **Route**, **Site Proxy**, or **Resolve and Route** actions.

Table 38 shows the Handle Client Error script parameters.

Table 38: Handle Client Error Script Parameters

Handle Client Error Script's Returned Value Type: Message	
Parameter	**Type**
event	RemoteNodeEvent
session	Session
requestFromServer	Message
requestToClient	Message
answerFromClient	Message
userTraceLogger	UserTraceLoggerWrapper
metaData	MetaData

Note: You may only call API methods associated with the parameters in *Table 38*. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar). This script is invoked when SDC routes an error message to a server peer (as the destination peer).

The following is an example of a **Handle Client Error** script:

```
return answerFromClient;
```

- **Handle Locally**

In **Handle Locally**, define a script to set if a message should be handled locally on an SDC site, and how it should be handled. This option is available when selecting **Forward**, **Route**, **Site Proxy**, actions.

Table 39 details the parameters SDC provides to the script:

Table 39: Handle Locally Script Parameters

Handle Locally Script's Returned Value Type: Boolean	
Parameter	**Type**
Session	Session

Handle Locally Script’s Returned Value Type: Boolean	
Stack	Stack
incomingMessage	Message
sourceRequest	Message
sourceAnswer	Message
sourcePeer	Peer
userTraceLogger	UserTraceLoggerWrapper
Metadata	MetaData

Note: You may only call API methods associated with the parameters in *Table 39*. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

Table 40 shows the parameters SDC provides to the script:

Table 40: Handle Locally Script Parameters

Handle Locally Script’s Returned Value Type: Message	
Parameter	**Type**
Session	Session
Stack	Stack
incomingMessage	Message
sourceRequest	Message
sourceAnswer	Message
sourcePeer	Peer
userTraceLogger	UserTraceLoggerWrapper
Metadata	MetaData

- **Redirect**

In **Redirect**, set the script to perform when **Redirect** Routing Action is selected.

Table 41 details the parameters SDC provides to the script:

Table 41: Redirect Script Parameters

Redirect Script's Returned Value Type: Message	
Parameter	**Type**
session	Session
Stack	Stack
envelope	Envelope
incomingMessage	Message
sourceRequest	Message
sourceAnswer	Message
sourcePeer	Peer
userTraceLogger	UserTraceLoggerWrapper
metaData	MetaData

Note: You may only call API methods associated with the parameters in *Table 41*. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

The following is an example of a **Redirect** script:

```
def answer = sourceRequest.createAnswer();
                                                                def redirectHostUsage
= envelope.getProperty("Redirect-Host-Usage");
                                                                if(redirectHostUsage!
= null){
      answer.add("Redirect-Host-Usage", redirectHostUsage);
            }
       def redirectHost = envelope.getProperty("Redirect-Host");
                                                                if(redirectHost !=
null){
       answer.add("Redirect-Host",
CodingUtils.asciiToBytes(redirectHost.toString()));
            }
          Long redirectMaxCacheTime = (Long)envelope.getProperty("Redirect-Max-
Cache-Time");
        if(redirectMaxCacheTime!= null){
         answer.add("Redirect-Max-Cache-Time", redirectMaxCacheTime);
            }
//answer.add("Redirect-Host", "redirect host name");
            return answer;
```

- **Handle Reject**

In **Handle Reject**, define a script to perform when a **Reject** Routing Action is selected.

Table 42 shows the parameters SDC provides to the script:

Table 42: Reject Script Parameters

Handle Reject Script's Returned Value Type: Message	
Parameter	**Type**
Session	Session
Stack	Stack
envelope	Envelope
incomingMessage	Message
sourceRequest	Message
sourceAnswer	Message
sourcePeer	Peer
userTraceLogger	UserTraceLoggerWrapper
metaData	MetaData

Note: You may only call API methods associated with the parameters in *Table 42*. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

The following is an example of a **Handle Reject** script:

```
Long resultCode = (Long)envelope.getProperty("Result-Code");

                                                                def answer =
sourceRequest.createAnswer(resultCode);


                                                                return answer;
```

- **Create Message Locally**

In **Create Message Locally**, define the exact way to create the local Message (local messages are returned to the Client Peer without having been forwarded to any Server Peer.

Table 43 shows the **Create Message Locally** script parameters.

Table 43: Create Message Locally Script Parameters

Create Answer Locally Script's Returned Value Type: Message	
Parameter	**Type**
Session	Session
sourceRequest	Message
sourceAnswer	Message
sourcePeer	Peer
userTraceLogger	UserTraceLoggerWrapper
Metadata	MetaData

Note: You may only call API methods associated with the parameters in *Table 43*. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

The following is an example of a Create Message Locally script:

```
def answer = sourceRequest.createAnswer(2001);
return answer;
```

4.7 Transformation

Note: While the functionalities described in this section can be configured in both SDC and EMS Web UI, it is recommended to perform these configurations globally using the EMS Web UI.

The Transformation decision tables defines how to apply transformation scripts to an incoming message before it is processed by the session management and routing rules decision tables and/or after it is processed by them.

4.7.1 Assigning Messages to a Transformation Script

1. **To create a new Transformation rule:**

2. Go to **Flows** > **Flows <Flows Name>** > **Transformation**. The Routing Rules screen is displayed.

3. Configure Rule Attributes for the Transformation rules by following the instructions in *Adding Rule Attributes.*

4. Define the rule criteria for the Transformation rules by following the instructions in *Defining the Rule Criteria*.

Under **Script**, type in the script to invoke when the conditions of the rule are met.

4.7.1.1 Adding a Transformation Script

You can add a script to be invoked when the conditions of a Transformation rule are met.

Table 44 shows the parameters that SDC provides to the script:

Table 44: Transformation Script Parameters

Transformation Condition Script's Returned Value Type: Message	
Parameter	**Type**
incomingMessage	Message
pendingIncomingRequest	Message
sourcePeer	Peer
destinationPeer	Peer
envelope	Envelope Note: The envelope is a data object that can be applied to pending requests. It contains concurrent hash map for the use of each transaction event (incoming/outgoing transformation).
userTraceLogger	UserTraceLoggerWrapper
metaData	MetaData

Note: You may only call API methods associated with the parameters in *Table 44*. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

The following is an example of a Transformation script:

```
Message copyOfRequest =
session.createRequest(incomingMessage);
copyOfRequest.removeAll(
"Accounting-Interim-Interval");
copyOfRequest.add(
"Accounting-Interim-Interval",99L); //unsigned32
//Update avp using set() method
copyOfRequest.add("User-Name","ScriptFlowTest1");
// Adding diameterIdentity
copyOfRequest.add(
"Destination-Host", "server2.traffix.com");
```

```
// Adding diameterIdentity
copyOfRequest.add(
"Destination-Realm", "traffix.com");
// Removing content
Content art = copyOfRequest.getValue(
"Accounting-Record-Type");
art.remove();
// Adding enumerated
copyOfRequest.add("Accounting-Record-Type", 3);
return copyOfRequest;
```

Click **Submit**.

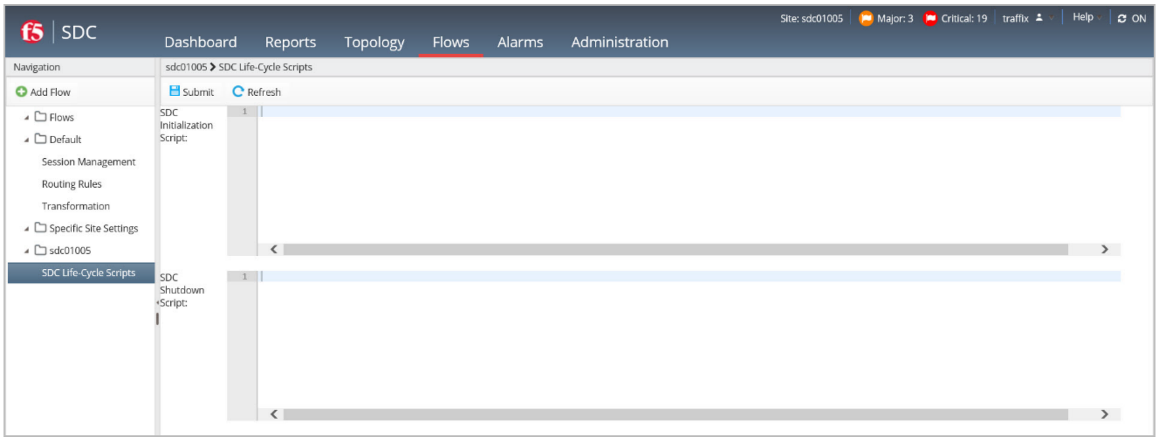
5.
4.8 SDC Life Cycle Scripts

As an SDC Web UI user, you may compose special scripts that run upon each SDC initialization and shutdown. The scripts may be used, for example, to load external table or database, load initial parameter values.

To implement the SDC Life Cycle script:

1. Go to **Flows** > **Flows** > **Specific Site Settings** > **SDC Life Cycle Scripts**. The SDC Life Cycle Scripts screen is displayed.

Figure 34: SDC Life Cycle Scripts



2. In **SDC Initialization Script** and **SDC Shutdown Script**, set the scripts to run each time an SDC is initiated or shuts down, respectively.

Table 45 shows the parameters SDC provides to the scripts:

Table 45: SDC Life Cycle Script Parameters

Parameter	Type
Stack	Stack
metaData	MetaData

Note: You may only call API methods associated with the parameters in *Table 45*. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

5. Monitoring the SDC

This chapter describes the different ways that you can monitor the SDC activity and performance.

5.1 Threshold Management

Threshold Management allows you to set the operational thresholds for alarm execution and KPI generation. Each category is assigned a critical, a major and a minor threshold. Alarms triggered by the system provide the severity threshold which caused their invocation.

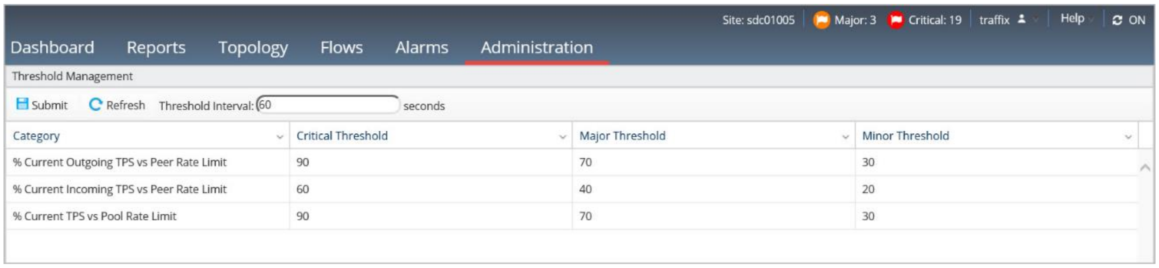
You may set severity thresholds to the following **Application Threshold** categories:

- Current Outgoing TPS vs Peer Rate Limit
- Current Incoming TPS vs Peer Rate Limit
- Current TPS vs Pool Rate Limit

To set the thresholds:

1. Go to **Administration** > **Threshold Management**. The Threshold Management screen is displayed.

Figure 35: Threshold Management



Threshold Interval: 60 seconds

Category	Critical Threshold	Major Threshold	Minor Threshold
% Current Outgoing TPS vs Peer Rate Limit	90	70	30
% Current Incoming TPS vs Peer Rate Limit	60	40	20
% Current TPS vs Pool Rate Limit	90	70	30

2.

3.

4. Select a category, and then set the **Critical**, **Major** and **Minor** thresholds.

In Threshold Interval, you can set the interval time (in seconds) for a threshold period.

Click **Submit**.

5.2 Dashboard

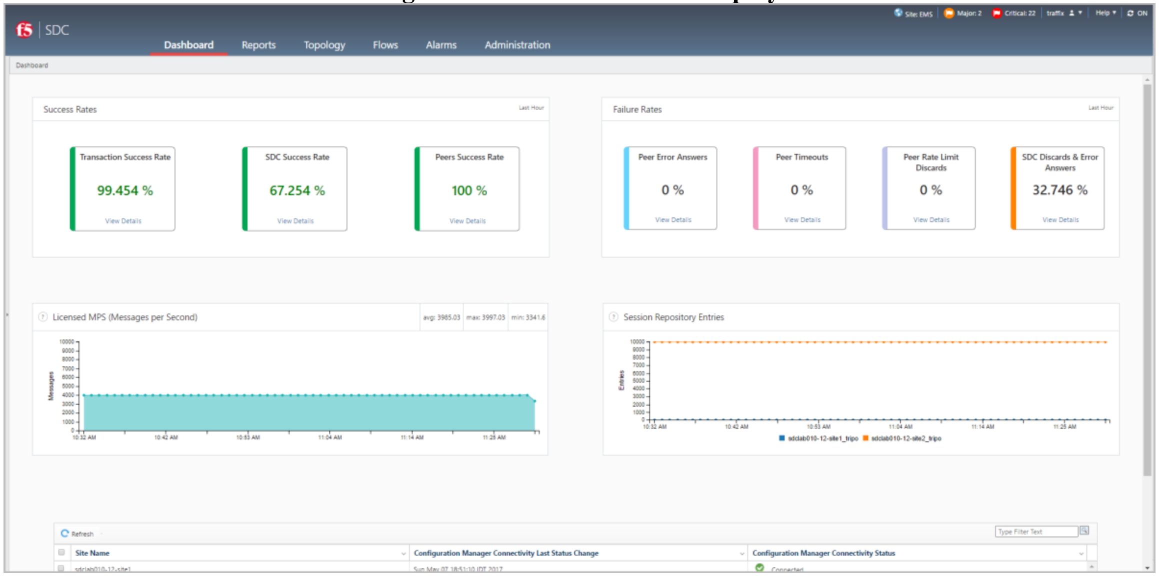
The Dashboard tab provides a centralized, high level view of message processing trends for the monitored site(s). This information is displayed for the last hour and is automatically refreshed every minute

To view the Dashboard:

From the tab menu in the EMS or SDC Web UI, click **Dashboard**.

1. The Dashboard screen is displayed, as depicted in *Figure 36*.

Figure 36: SDC Dashboard Display



Site Name	Configuration Manager Connectivity Last Status Change	Configuration Manager Connectivity Status
sdclab010-12-site1	Sun May 07 18:51:10 IDT 2017	✓ Connected

Table 46 details and describes the Dashboard graphs.

Note: Due to the data processing time, the information is presented with a delay of approximately 2 minutes.

Table 46: Dashboard Graphs

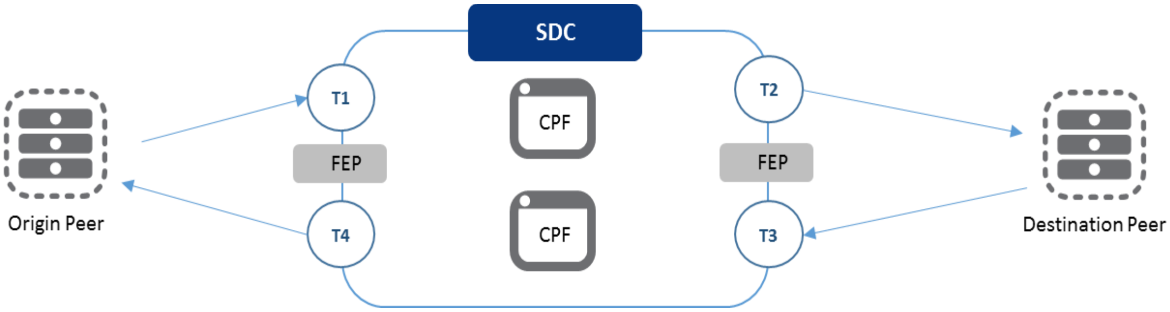
Graph	Description
Success Rates	

Graph	Description
Transactions Success Rate	The percentage of transaction requests received by the SDC from the origin peer that were returned to the origin peer by the SDC with a success answer (protocol-based success result code)
SDC Success Rate	The percentage of transaction requests received by the SDC from the origin peer that were successfully processed, and were returned to the origin peer by the SDC without any SDC-based errors.
Peers Success Rate	The percentage of answers returned to the SDC from the destination peer with a protocol-based success result code.
Failure Rates	
Peer Error Answers	The percentage of answers returned to the SDC from the destination peer with a protocol-based error result code.
Peer Timeouts	The percentage of requests that timed out and were not processed by the peer.
Peer Rate Limit Discards	The percentage of requests that were rejected by the SDC since they exceeded the incoming rate limit defined for the peer that they originated from.
SDC Discards & Error Answers	The percentage of requests and answers that were discarded, and the number of answers returned to the peer from the SDC with an SDC-based error answer.
Licensed MPS (Messages per Second)	The total number of incoming requests, outgoing requests, incoming answers and outgoing answers sent between the SDC and its connected peers. Messages exchanged between two SDC sites, as well as peer connection establishment messages, are not counted.
Session Repository Entries	The total number of entries managed by the SDC Session Repository (Tripo).

5.3 Reports

The Reports tab displays the performance data collected by the SDC for the basic traffic flow between the SDC and its connected peers as illustrated in *Figure 37*.

Figure 37: Message Transaction Flow



In this flow, message requests are sent from the origin peer to the SDC (T1). These message requests are then processed by the SDC and sent to the destination peer (T2). Message answers are sent back from the destination peer to the SDC (T3). These message answers are then processed by the SDC and sent to the origin peer (T4).

5.3.1 Time Resolution

The collected data can be presented in the Reports screens in different time resolutions, as follows:

- Hour – the data is presented for the last hour, in one minute increments.
- Day – the data is presented for the last 24 hours, in 15 minute increments.
- Week – the data is presented for the last week, in hour increments.
- Month – the data is presented for the last month, in 4 hour increments.
- Year – the data is presented for the last year, in 24 hour increments.

To define the time resolution that the collected data is displayed in:

1. Go to the **Reports** tab and navigate to the desired screen.
2. Select the desired Time Resolution from the screen title bar.

Note: In the Time Resolution table, the resolution is defined based on the last rounded time period, while in the graphs below the table, the resolution is defined based on the actual last time period.

Auto-refresh is done per the defined increment per time resolution. For example, for a day resolution, auto-refresh is done every 15 minutes.

5.3.2 Transaction Related KPIs

5.3.2.1 Transactions Summary

The Transaction Summary screen provides an overview of how the SDC and its connected peers are processing received transactions. This overview is provided from three perspectives, by three types of reports:

- The average per second reports – each report shows the average value per second in the monitored interval.
- The over time trends graphs – each graph shows the incremental data for the monitored period, according to the selected time resolution. For more information, see *Time Resolution*.
- The breakdown pie charts – each pie chart shows the overall data for the monitored period.

1.

To view the average per second reports:

From the tab menu click **Reports** > **Transactions** > **Transactions Summary**.

The average per second reports are displayed at the top of the screen. *Table 47* details the provided reports.

Table 47: Transactions Summary - Average per Second Reports

Report	Description
Requests Received From Peers	The number of request messages received from the connected peers per second.

Report	Description
Transaction Successful Answers	The number of transactions that were returned to the origin peer with a protocol-based success result code per second.
SDC Successes	The number of transaction requests received by the SDC from the origin peer that were successfully processed, and were then returned to the origin peer by the SDC without any SDC-based errors per second.
Peers Successful Answers	The number of transaction responses returned to the SDC from the destination peer with a protocol-based success result code per second.
SDC Errors & Discards	The number of SDC-generated error responses and discards per second. In the related graphs, these errors and discards are broken down into the following errors: ▪ No_Destination_Found – the SDC could not find an available peer to send the request to. ▪ Process_Rate_Limit_Exceeded – the SDC could not process the request because the SDC had reached its defined rate limit. ▪ SDC_Application_Error – the SDC could not process the request due to an SDC application-based error. For example, a CPF-based error or a FEP-based error. ▪ SDC_Overload – the SDC could not process the request because an internal queue was full. ▪ User_Input_Error- the SDC could not process the request, either because it did not match any routing rule, or an error occurred when invoking one of the processing scripts. The discard reasons are: ▪ Missing Routing Discards – the SDC could not find a routing rule to match the request. ▪ SDC Overloads Discards – one of the SDC processing queues was full and could not process the request.

Report	Description
Peers Errors Answers	The number of error answers returned by the destination peer per second. In the related graphs, these errors are presented as "families of remote node events" as follows: Note: These categories reflect Diameter result codes. ▪ Permanent_Failure – the number of peer responses with a result code of 5XXX. ▪ Transient_Failure – the number of peer responses with a result code of 4XXX. ▪ Protocol_Error – the number of peer responses with a result code of 3XXX. Note: Protocol_Error for HTTP reflects the 4XX and 5XX result codes.
Peer Timeouts	The number of requests that timed out waiting for a response from the destination peer per second.

The over time trends graphs and breakdown pie charts are displayed below the average per second reports, by category. Each over time trends graph has a corresponding breakdown pie chart, further expanding on the trend.

1.

To view the over time trends graphs and breakdown pie charts:

From the tab menu click **Reports** > **Transactions** > **Transactions Summary**.

The over time trends graphs and breakdown pie charts are displayed by category. *Table 48* details the over time trends graphs and their corresponding breakdown pie chart. To temporarily focus on one specific trend in an over time trends graph, hover over the trend name on the legend at the bottom of the graph. To permanently focus on one specific trend, click the trend name that you want to filter off the graph.

Each category displays the collected information for all transactions by transaction type. To view information for a specific transaction type, select the desired transaction type from the drop-down list at the right of each category heading.

Some over time trends graphs also display the average, maximum, and minimum value recorded in the monitored time interval. These values are displayed to the top right of the specific graph.

Table 48: Transactions Summary – Over Time Trends Graphs & Breakdown Pie Charts by Category

Name	Description
Transaction Successful Answers vs Failures	
Transaction Successful Answers vs Failures	The number of transaction requests that were processed by a destination peer and returned to the origin peer with a success answer, as well as the number of transaction requests that were returned to the origin peer with either a peer-based or SDC-based error answer, discard, or timeout.
Transaction Success vs Failure Rates	The overall number of transaction requests returned to the origin peer, broken down into either success answers or failures.
SDC Successes	
SDC Successes	The number of transaction requests received by the SDC from the origin peer that were successfully processed, and were returned to the origin peer by the SDC without any SDC-based errors.
SDC Success vs Discard & Error Answer Rates	The overall number of transaction requests returned to the origin peer, broken down into either SDC-based success answers, or SDC-based error answers and discards.

Name	Description
Peers Successful Answers	
Peers Successful Answers	Number of transaction requests processed by the destination peer and returned to the SDC from the destination peer with a protocol-based success result code.
Peer Success vs Failure Answer Rates	The overall number of transaction requests processed by the destination peer and returned to the SDC, broken down into either peer-based success answers, or peer-based error answers and timeouts.
SDC Errors & Discards	
SDC Errors & Discards	Number of requests and answers that were discarded, and the number of answers returned to the origin peer from the SDC with an SDC-based error answer.
SDC Errors & Discards Reasons	The overall number of requests and answers that were discarded, broken down into the specific SDC-based errors.
Peer Error Answers	
Peer Error Answers	Number of answers returned to the SDC from the destination peer with a protocol-based error result code.
Peer Error Reasons	The overall number of error answers returned to the SDC from the destination peer, broken down into the specific protocol-based errors.
Peer Timeouts	
Peer Timeouts	Number of requests that timed out and were not processed by the peer.

Name	Description
Errors vs Timeouts	The overall number of error answers returned to the SDC from the destination peer, broken down into protocol-based errors or time outs.

5.3.3 SDC Related KPIs

5.3.3.1 Summary

The Summary screen provides a high-level view of the site behavior, including traffic processing, connected peers, and Session Repository traffic.

This overview is provided from two perspectives, by two types of reports:

- The maximum per selected time resolution reports - each report shows the highest recorded incremental value in the monitored interval between the last time_slot and the current time_slot. The increment is defined according to the selected time resolution. For more information, see *Time Resolution*.
- The over time trends graphs – each graph shows the incremental data for the monitored period, according to the selected time resolution. For more information, see *Time Resolution*. To temporarily focus on one specific trend in an over time trends graph, hover over the trend name on the legend at the bottom of the graph. To permanently focus on one specific trend, click the trend name that you want to filter off the graph.

1.

To view the SDC Summary data:

From the tab menu click **Reports** > **SDC** > **Summary**.

The maximum per selected time resolution reports are displayed at the top of the screen, above the over time trends graphs for the same data. *Table 49* details the SDC Summary data.

Table 49: SDC Summary Data

Report	Description
Licensed MPS	The sum of all the incoming requests, outgoing requests, incoming answers and outgoing answers between the SDC site and its connected peers. Messages exchanged between two SDC sites, and peer connection establishment messages, are not counted. Note: The overtime trends graph displays the collected information for all CPFs. To view information for a specific CPF, select the desired CPF from the drop-down list at the right of the graph heading.
Processed MPS	The sum of all the incoming requests, outgoing requests, incoming answers and outgoing answers between the SDC site and its connected peers. Messages exchanged between two SDC sites, and peer connection establishment messages, are also counted. Note: The over time trends graph displays the collected information for all CPFs. To view information for a specific CPF, select the desired CPF from the drop-down list at the right of the graph heading.
Connected Peers	The recorded number of peers connected to the SDC site during the monitored interval.
Session Repository Entries	The recorded number of entries in the Session Repository during the monitored interval.

5.3.3.2 SDC Queues

The SDC Queues screen provides an overview of the site traffic processing behavior.

This overview is provided from two perspectives, by two types of reports:

- The maximum per selected time resolution reports - each report shows the highest recorded incremental value in the monitored interval between the last time_slot and the current time_slot. The increment is defined according to the selected time resolution. For more information, see *Time Resolution*.

- The over time trends graphs – each graph shows the incremental data for the monitored period, according to the selected time resolution. For more information, see *Time Resolution*. To temporarily focus on one specific trend in an over time trends graph, hover over the trend name on the legend at the bottom of the graph. To permanently focus on one specific trend, click the trend name that you want to filter off the graph.

To view the SDC Queues data:

1. From the tab menu click **Reports > SDC > SDC Queues**.

The maximum per selected time resolution reports are displayed at the top of the screen, above the over time trends graphs for the same data. *Table 50* details the SDC Queues data.

Table 50: SDC Queues Data

Collected Information	Description
Incoming Requests Queue Usage	The number of incoming requests in the SDC requests queue.
Incoming Answers Queue Usage	The number of incoming answers in the SDC answers queue.
Diameter Pending Requests Queue Usage	The number of requests waiting for an answer from the destination peer.
Session Repository Queue Usage	The number of incoming requests in the session repository incoming requests queue.

5.3.3.3 Session Repository

The Session Repository screen provides an overview of the Session Repository behavior, including the used capacity, entry creation attempts, and more.

This overview is provided from two perspectives, by two types of reports:

- The maximum per selected time resolution reports - each report shows the highest recorded incremental value in the monitored interval between the last time_slot and the current time_slot. The increment is defined according to the selected time resolution. For more information, see *Time Resolution*.
- The over time trends graphs – each graph shows the incremental data for the monitored period, according to the selected time resolution. For more information, see *Time Resolution.* To temporarily focus on one specific trend in an over time trends graph, hover over the trend name on the legend at the bottom of the graph. To permanently focus on one specific trend, click the trend name that you want to filter off the graph.

To view the Session Repository data:

1. From the tab menu click **Reports** > **SDC** > **Session Repository**.

The maximum per selected time resolution reports are displayed at the top of the screen, above the over time trends graphs for the same data. *Table 51* details the Session Repository data.

Note: The data is displayed in two tables. Select **KPIs** or **More** to view the relevant data.

Table 51: Session Repository Data

Collected Information	Description
KPIs	
Session Repository Entries	The recorded number of entries in the Session Repository during the monitored interval.
Successful Session Bindings	The number of successful session bindings per SDC site during the previous measurement period.

Collected Information	Description
	Slave session transactions are also counted if the binding to the master was successful, since every transaction of the slave session is bound to the master session.
Failed Session Bindings	The number of failed session bindings per SDC site during the previous measurement period. Secondary session transactions are also counted if binding to master failed.
Expired Entries	The number of sessions that expired per SDC site during the previous measurement period.
Deleted Entries	The number of Session Repository entries that were deleted per SDC site during the previous measurement period.
More	
Successful Addition Attempts	The number of new Session Repository entries created during the previous measurement period. Each replicated entry is only counted once
Failed Addition Attempts	The number of failed new Session Repository entry attempts during the previous measurement period
Expired Entries	The number of expired sessions during the previous measurement period
Deleted Entries	The number of deleted sessions during the previous measurement period
Failed Attempts to Delete Entry	The number of failed attempts to delete Session Repository entries during the previous measurement period. Includes attempts to delete entries that did not exist.
Sent SRRs	The number of replication requests that were sent to the geo-redundant site.
Received SRRs	The number of replication requests that were received from the geo-redundant site.

Collected Information	Description
Successful Session Repository Queries	The number of session repository queries that successfully returned a session repository entry value.
Failed Session Repository Queries	The number of session repository queries that failed to return a session repository entry value.

5.3.3.4 Routing Row Requests

The Routing Row Requests screen provides an overview of request routing trends based on a specific row in the Routing Rule decision table.

To view the Routing Row Requests Data:

1. From the tab menu click **Reports** > **SDC** > **Routing Row Requests**.

Table 52 details and describes the Routing Row Requests table columns.

Table 52: Routing Row Requests Table Columns

Graph	Description
Site/Routing ID	The name of the monitored site and the ID of the routing row.
Successful Routing Attempts	The number of requests that matched the routing row rule criteria and were successfully routed to the destination peer.
Failed Routing Attempts	The number of requests that matched the routing row rule criteria and did not successfully route to the destination peer.

2. Click on a row to open graphs corresponding to each of the table columns. Each graph shows the collected data over time to provide a view of the behavioral trend for each type of collected data.

5.3.4 Latency Related KPIs

5.3.4.1 Latency Summary

The Latency Summary screen provides an overview of the latency the site is experiencing.

This overview is provided from two perspectives, by two types of reports:

- The maximum per selected time resolution reports - each report shows the highest recorded incremental value in the monitored interval between the last time_slot and the current time_slot The increment is defined according to the selected time resolution. For more information, see *Time Resolution*.
- The over time trends graphs – each graph shows the incremental data for the monitored period, according to the selected time resolution. For more information, see *Time Resolution.* To temporarily focus on one specific trend in an over time trends graph, hover over the trend name on the legend at the bottom of the graph. To permanently focus on one specific trend, click the trend name that you want to filter off the graph.

To view the Latency Summary data:

1. From the tab menu click **Reports** > **Latency** > **Latency Summary**.

The maximum per selected time resolution reports are displayed at the top of the screen, above the over time trends graphs for the same data. *Table 53* details the Latency Summary data.

Table 53: Latency Summary Data

Collected Information	Description
Transaction End to End Latency	The average roundtrip time of transactions handled by the SDC. It is measured from the moment a request is received by the SDC until the answer is sent to the peer that initiated the transaction
SDC Request Latency	The average latency of requests handled by the SDC. It is measured from the moment a request is received by the SDC until the request is sent to the designated destination.
SDC Answer Latency	The average latency of answers handled by the SDC. It is measured from the moment an answer is received by the SDC until the answer is sent to the remote peer that initiated the transaction.

5.3.5 Peer Related KPIs

5.3.5.1 Peers Health

The Peers Health screen provides an overview of the peer health parameters. The health parameters are identical to those that are used to calculate a peer's health (**Topology** > **Peers** > **Health**). When a parameter value exceeds the defined threshold, it is displayed in red in the table and above a red line in the graphs.

To view the Peers Health Data:

From the tab menu click **Reports > Peers > Peers Health**.

1. *Table 54* details and describes the Peers Health table columns.

Table 54: Peers Health Table Columns

Graph	Description
Peer	The name of the connected peer that data is presented for.
Error Answers Ratio (%)	The ratio between the number of requests sent to the destination peer and the number of error answers returned from the peer.
Error Answers (per sec)	The number of error answers sent to the SDC by the destination peer
Timeout Ratio (%)	The ratio between the number of requests sent to the destination peer and the number of requests that were not answered by the destination peer
Timeout (per sec)	Number of requests sent by the SDC that were not answered by the destination peer
Round Trip Time (per sec)	The average processing time of requests by the destination peer.
Requests Received from Peer (per sec)	Number of requests received by the SDC from a destination peer.
Requests Sent to Peer (per sec)	Number of requests sent by the SDC to a destination peer
Network Write Queue Usage (per sec)	The SDC queue of answers and requests that are waiting to be written to machine socket

Graph	Description
Out of Service Time (%)	The percentage of time that the destination peer was out of service.

2. Click on a row to open graphs corresponding to each of the table columns. Each graph shows the collected data over time to provide a view of the behavioral trend for each type of collected data.

5.3.5.2 Traffic Returned From Peer

The Traffic Returned From Peer screen provides an overview of traffic returning from the destination peers to the SDC site.

To view the Traffic Returned From Peer Data:

1. From the tab menu click **Reports > Peers > Traffic Returned From Peer**.

Table 55 details and describes the Traffic Returned From Peer table columns.

The data is displayed in two tables. Select **Summary** or **Peer Rate Limit** to view the relevant data.

Table 55: Traffic Returned From Peer Table Columns

Graph	Description
Summary	
Peer	The name of the destination peer that data is presented for.
Site (EMS only)	The name of the monitored site that data is presented for.
Requests Sent to Peer (per sec)	The number of requests sent by the SDC to a destination peer.
Answers (per sec)	The average number of answers (both success and error) returned from the destination peer to the SDC.
Successful Answers (per sec)	Number of success answers returned from the destination peer to the SDC.
Error Answers (per sec)	The average number of error answers returned from the destination peer to the SDC..

Graph	Description
Timeouts (per sec)	The average number of requests sent by the SDC that were not answered by the destination peer.
Round Trip Time (per sec)	The average processing time (in milliseconds) of requests by the destination peer. Measured from when the request is received by the destination peer until the answer is sent back to the SDC.
99.9 Percentile Round Trip Time (per sec)	The average processing time (in milliseconds) of 99.9 percent of the requests by the destination peer.
Peer Rate Limit	
Requests Received from Peer Before Rate Limit (per sec)	Number of requests received by the SDC from the origin peer, without going over the rate limit. Note: Rate limits can be user defined or defined by default.
Requests Received from Peer (per sec)	Number of requests received by the SDC from the origin peer, including requests rejected by the SDC due to an exceeded rate limit.
Requests Rejects by Peer Rate Limit (per sec)	The number of requests which were rejected due to going over the rate limit. Note: Rate limits can be user defined or defined by default.

2. Click on a row to open graphs corresponding to each of the table columns. Each graph shows the collected data over time to provide a view of the behavioral trend for each type of collected data.

5.3.5.3 Traffic Returned to Peer

The Traffic Returned to Peer screen provides an overview of the traffic that is returned to the origin peer after being processed by the SDC site.

To view the Traffic Returned to Peer Data:

1. From the tab menu click **Reports > Peers > Traffic Returned to Peer**

Table 56 details and describes the Traffic Returned to Peer table columns.

Table 56: Traffic Returned to Peer Table Columns

Graph	Description
Site (EMS only)	The name of the monitored site that data is presented for.
Peer	The name of the origin peer that data is presented for.
Requests Received from Peer (per sec)	Number of requests received by the SDC from a destination peer.
Successful Answers (per sec)	The number of requests that were returned to the origin peer with a protocol-based success result code
SDC Received Errors & Discards (per sec)	The total number of requests that were returned to the origin peer with an SDC based error or were discarded by the SDC.
Peer Received Error Answers (per sec)	The total number of requests that were returned to the origin peer with a destination peer based error In the related graphs, these errors are presented as "families of remote node events" as follows: Note: These categories reflect Diameter result codes. ▪ Permanent_Failure – the number of peer responses with a result code of 5XXX. ▪ Transient_Failure – the number of peer responses with a result code of 4XXX. ▪ Protocol_Error – the number of peer responses with a result code of 3XXX. Note: Protocol_Error for HTTP reflects the 4XX and 5XX result codes.
Peer Received Timeouts (per sec)	The total number of timed out requests received by the origin peer. A timed out request is the result of what happened while trying to be processed by the destination peer.

Click on a row to open graphs corresponding to each of the table columns. Each graph shows the collected data over time to provide a view of the behavioral trend for each type of collected data.

2.

5.3.5.4 Traffic by Bytes

The Traffic by Bytes screen provides an overview of traffic by bytes sent to and received from peers.

To view the Traffic by Bytes Data:

From the tab menu click **Reports** > **Peers** > **Traffic by Bytes**.

1. Table 57 details and describes the Traffic by Bytes table columns.

Table 57: Traffic by Bytes Table Columns

Graph	Description
Site (EMS only)	The name of the monitored site that data is presented for.
Peer	The name of the origin peer that data is presented for.
Bytes Requests Received from Peer (per sec.)	Number of bytes in requests received by the SDC from the origin peer.
Bytes Answers Sent to Origin Peer (per sec.)	Number of bytes in answers sent by the SDC to the origin peer
Bytes Requests Sent to Peer (per sec.)	Number of bytes in requests sent by the SDC to the destination peer.
Bytes Answers Received from Destination Peer (per sec.)	Number of bytes in answers sent to the SDC from the destination peer

Click on a row to open graphs corresponding to each of the table columns. Each graph shows the collected data over time to provide a view of the behavioral trend for each type of collected data

5.3.6 Pool Related KPIs

5.3.6.1 Pools Health

The Pool Health screen provides an overview of the pool health parameters. The health parameters are identical to those that are used to calculate a pool's health (**Topology** > **Pools** > **Health**). When a parameter value exceeds the defined threshold, it is displayed in red in the table and above a red line in the graphs.

To view the Pools Health Data:

From the tab menu click **Reports > Pools > Pools Health**.

1. *Table 58* details and describes the Pools Health table columns.

Table 58: Pools Health Table Columns

Graph	Description
Pool	The name of the pool of destination peers that data is presented for.
Error Answers Ratio (%)	The ratio between the number of requests sent to the pool of destination peers and the number of error answers returned from the pool of destination peers.
Error Answers (per sec)	The average number of error answers returned to the SDC by the pool of destination peers
Timeout Ratio (ms))	The ratio between the number of requests sent to the pool of destination peers and the number of requests that were not answered by the pool of destination peers.
Timeout (per sec)	Number of requests sent by the SDC that were not answered by the pool of destination peers.
Requests Sent to Pool (per sec)	Number of requests sent by the SDC to a specific pool of destination peers.

2. Click on a row to open graphs corresponding to each of the table columns. Each graph shows the collected data over time to provide a view of the behavioral trend for each type of collected data.

5.3.6.2 Traffic Returned From Pool

The Traffic Returned From Pool screen provides an overview of the incoming traffic trends and how traffic returning from the destination peers in a specific pool is processed by the site.

To view the Traffic Returned From Pool:

From the tab menu click **Reports > Peers > Traffic Returned From Pool**.

1. *Table 59* details and describes the Traffic Returned From Pool table columns.

Table 59: Traffic Returned From Pool Table Columns

Graph	Description
Pool	The name of the pool of destination peers that data is presented for.
Successful Answers (per sec)	The average number of success answers sent to the SDC by the pool of destination peers.
Error Answers (per sec)	The average number of error answers returned to the SDC by the pool of destination peers.
Timeouts (per sec)	The average number of requests sent by the SDC that were not answered by the pool of destination peers.
Round Trip Time (per sec)	The average processing time of requests by the destination peers in the pool.
99.9 Percentile Round Trip Time (v)	The average processing time of 99.9 percent of the requests sent to the pool of destination peers.

2.

Click on a row to open graphs corresponding to each of the table columns. Each graph shows the collected data over time to provide a view of the behavioral trend for each type of collected data.

5.3.7 Resource Related KPIs

5.3.7.1 Machine Summary

The Machine Summary screen provides a real-time view of the hosts’ performance.

To view the host performance data:

1. From the tab menu click **Reports** > **Resources** > **Machine Summary**.

Table 60 details and describes the columns of the Machine Summary table.

Table 60: Machine Summary Table Columns

Column	Description
Machine	The name of the SDC or EMS site that the data is presented for.
CPU Usage (%)	The percentage of CPU used by the host running the SDC component.
Free Memory (Gigabytes)	The amount of available memory in GB.
Machine Available Swap (Gigabytes)	The amount of available swap space in GB.
Machine Load Average	The ratio between the Operating System's load average counter and the number of CPU cores.

2. Click on a row to open graphs corresponding to each of the table columns. Each graph shows the collected data over time to provide a view of the behavioral trend for each type of collected data.

5.3.7.2 Network Usage

The Network Usage screen provides a real-time view of the network usage per component.

To view the network usage data:

1. From the tab menu click **Reports** > **Resources** > **Network Usage**.

Table 61 details and describes the columns of the Network Usage table.

Table 61: Network Usage Columns

Column	Description
Machine	The name of the SDC or EMS site that the data is presented for.
Sent Bytes (Mega Bytes)	The number of bytes sent, counted per component.

Column	Description
Received Bytes (Mega Bytes)	The amount of bytes received, counted per component.
Input Errors	The number of input errors, counted per component.
Output Errors	The number of output errors, counted per component.

Click on a row to open graphs corresponding to each of the table columns. Each graph shows the collected data over time to provide a view of the behavioral trend for each type of collected data.

2. From the **All** drop-down field, select the relevant eth interface for which you want to view the collected data.

3.

5.3.7.3 SDC Components

The SDC Components screen provides a real-time view of the FEP and CPF components performance.

To view the SDC Components performance data:

1. From the tab menu click **Reports** > **Resources** > **SDC Components**.

Table 62 details and describes the columns of the Summary table.

Table 62: SDC Components Table Columns

Column	Description
Node	The name of the site that the data is presented for.
CPU Usage (%)	The percentage of CPU used by the host running the CPF and FEP components.
Used Memory	The memory (in MB) that the CPF and FEP components consumed.

2. Click on a row to open graphs corresponding to each of the table columns. Each graph shows the collected data over time to provide a view of the behavioral trend for each type of collected data.

5.3.8 Transaction Record Data (TDR) Related KPIs

5.3.8.1 TDR Dashboard

The TDR Dashboard displays a graph of the Top 10 Origin-Destination channels per category in the selected time frame. The information displayed in the TDR Dashboard reflects one of the following five categories:

- Number of Messages
- Round Trip Time
- OK Responses
- Timeouts
- Other Errors

Once a category is selected, the TDR Dashboard displays the Top 10 Origin-Destination channels for the selected category in the selected time frame.

5.3.8.2 Transactions Data Records

The Transaction Data Reports screen displays all the system TDRs. TDRs can be filtered by one or more of four predefined common TDR fields (Origin Realm, Origin Host, Destination Realm, and Destination Host.), or by a user-defined filter.

Table 63 details the collected data in each generated TDR.

Note: All data fields should be written with quotations marks (") to avoid data file format issues.

TDR reports are also exported as tdr_export_<date>_<time>.csv.gz files. For sites that are already updated with ELK, TDR reports are sent to the /opt/traffix/reports/elk folder of the EMS site, and for those sites that have not yet been updated (going through the update Splunk-ELK process), TDR reports are sent to the /opt/traffix/reports/tdr folder of the EMS site. The default setting is that the three most recently generated TDR .csv files are exported every 30 minutes. Refer to the *F5 SDC Troubleshooting Guide* for information on how to configure how often reports should be generated, how many should be saved, the file name format, and the export location of the reports.

Table 63: TDR Collected Data

Data Field	Data Type	Description
_time	Timestamp	The timestamp of the transaction.
Origin_Realm	String	Realm where the incoming request originated from.
Origin_Host	String	The peer name from which the request was received.
Destination_Realm	String	Destination realm of the request, taken from the incoming request.
Destination_Host	String	The peer name the request is sent to.
CMD_Code	String	Command code of every interface taken from the incoming request. For example ULR, CCR.
Result_Code	Integer	The result code of the transaction.
Origin_Host_Request	String	The Origin Host extracted from the incoming request’s AVP.
Origin_Host_Answer	String	The Origin Host extracted from the incoming answer’s AVP.
Diameter_Result_Code	Integer	The result code that is sent to the transaction originator, taken from the outgoing response.
IMSI	Numeric String	The subscriber identifier, taken from the incoming request.
Roundtrip_Time	Milliseconds	The time in milliseconds from when the request was sent to the transaction destination peer until a response was received.

Data Field	Data Type	Description
Source_Application_Id	Integer	Application ID from the original incoming request.
Destination_Application_Id	Integer	Application ID from the outgoing request.
Destination_Command_Code	Integer	Command code of the transaction, taken from the outgoing request.
Flow_Total_Time	Milliseconds	The milliseconds that passed once the request was received by the SDC and a response was sent back to the originator.
Original_Request_Length	Numeric String	The length of the original request message.
Sending_Request_Length	Numeric String	The length of the outgoing request message.
Original_Response_Length	Numeric String	The length of the original response message.
Answer_To_Client_Length	Numeric String	The length of the outgoing response message.
Original_Result_Code	Numeric String	The result code from the incoming response.
AVP_1	User-defined	An additional AVP to be defined by the user.
AVP_2	User-defined	An additional AVP to be defined by the user.
AVP_3	User-defined	An additional AVP to be defined by the user.
AVP_4	User-defined	An additional AVP to be defined by the user.
AVP_5	User-defined	An additional AVP to be defined by the user.

5.3.8.3 Traced Messages

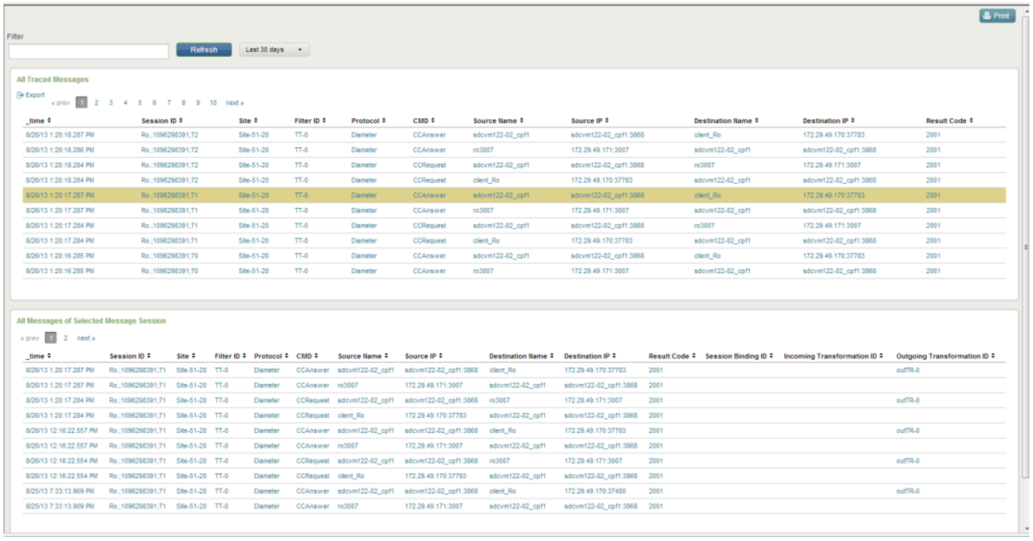
The Traced Messages displays a log of transactions made in your system.

1. Note: To activate message tracing see *Configuring a Tracing Rule.*

To view traced messages:

Go to **Reports** > **TDRs** > **Traced Messages**. The **Traced Messages** screen is displayed.

Figure 38: Traced Messages



All Traced Messages

_time	Session ID	Site	Filter ID	Protocol	CMD	Source Name	Source IP	Destination Name	Destination IP	Result Code
8/26/13 1:20:18.287 PM	Ro.;1096298391;72	Site-51-20	TT-0	Diameter	CCAnswer	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	client_Ro	172.29.49.170:37783	2001
8/26/13 1:20:18.286 PM	Ro.;1096298391;72	Site-51-20	TT-0	Diameter	CCAnswer	ro3007	172.29.49.171:3007	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	2001
8/26/13 1:20:18.284 PM	Ro.;1096298391;72	Site-51-20	TT-0	Diameter	CCRequest	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	ro3007	172.29.49.171:3007	2001
8/26/13 1:20:18.284 PM	Ro.;1096298391;72	Site-51-20	TT-0	Diameter	CCRequest	client_Ro	172.29.49.170:37783	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	2001
8/26/13 1:20:17.287 PM	Ro.;1096298391;71	Site-51-20	TT-0	Diameter	CCAnswer	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	client_Ro	172.29.49.170:37783	2001
8/26/13 1:20:17.287 PM	Ro.;1096298391;71	Site-51-20	TT-0	Diameter	CCAnswer	ro3007	172.29.49.171:3007	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	2001
8/26/13 1:20:17.284 PM	Ro.;1096298391;71	Site-51-20	TT-0	Diameter	CCRequest	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	ro3007	172.29.49.171:3007	2001
8/26/13 1:20:17.284 PM	Ro.;1096298391;71	Site-51-20	TT-0	Diameter	CCRequest	client_Ro	172.29.49.170:37783	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	2001
8/26/13 1:20:16.285 PM	Ro.;1096298391;70	Site-51-20	TT-0	Diameter	CCAnswer	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	client_Ro	172.29.49.170:37783	2001
8/26/13 1:20:16.285 PM	Ro.;1096298391;70	Site-51-20	TT-0	Diameter	CCAnswer	ro3007	172.29.49.171:3007	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	2001

All Messages of Selected Message Session

_time	Session ID	Site	Filter ID	Protocol	CMD	Source Name	Source IP	Destination Name	Destination IP	Result Code	Session Binding ID	Incoming Transformation ID	Outgoing Transformation ID
8/26/13 1:20:17.287 PM	Ro.;1096298391;71	Site-51-20	TT-0	Diameter	CCAnswer	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	client_Ro	172.29.49.170:37783	2001			outTR-0
8/26/13 1:20:17.287 PM	Ro.;1096298391;71	Site-51-20	TT-0	Diameter	CCAnswer	ro3007	172.29.49.171:3007	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	2001			
8/26/13 1:20:17.284 PM	Ro.;1096298391;71	Site-51-20	TT-0	Diameter	CCRequest	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	ro3007	172.29.49.171:3007	2001			outTR-0
8/26/13 1:20:17.284 PM	Ro.;1096298391;71	Site-51-20	TT-0	Diameter	CCRequest	client_Ro	172.29.49.170:37783	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	2001			
8/26/13 12:16:22.557 PM	Ro.;1096298391;71	Site-51-20	TT-0	Diameter	CCAnswer	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	client_Ro	172.29.49.170:37783	2001			outTR-0
8/26/13 12:16:22.557 PM	Ro.;1096298391;71	Site-51-20	TT-0	Diameter	CCAnswer	ro3007	172.29.49.171:3007	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	2001			
8/26/13 12:16:22.554 PM	Ro.;1096298391;71	Site-51-20	TT-0	Diameter	CCRequest	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	ro3007	172.29.49.171:3007	2001			outTR-0
8/26/13 12:16:22.554 PM	Ro.;1096298391;71	Site-51-20	TT-0	Diameter	CCRequest	client_Ro	172.29.49.170:37783	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	2001			
8/25/13 7:33:13.909 PM	Ro.;1096298391;71	Site-51-20	TT-0	Diameter	CCAnswer	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	client_Ro	172.29.49.170:37480	2001			outTR-0
8/25/13 7:33:13.909 PM	Ro.;1096298391;71	Site-51-20	TT-0	Diameter	CCAnswer	ro3007	172.29.49.171:3007	sdcvm122-02_cpf1	sdcvm122-02_cpf1:3868	2001			

The list displays a message log of transactions made in your system, and their properties: Session ID, Site, Filter ID, Protocol, CMD, Source Name and IP, Destination Name and IP, Result Code.

Each transaction is comprised of four messages:

- A request sent from the Client Peer to the SDC
- A request sent from SDC to the Server Peer
- An answer sent from the Server Peer to SDC
- An answer sent from SDC to the Client Peer

Clicking each message's line reveals the three other messages that are were involved in the transaction. Each message is detailed, in the table below.

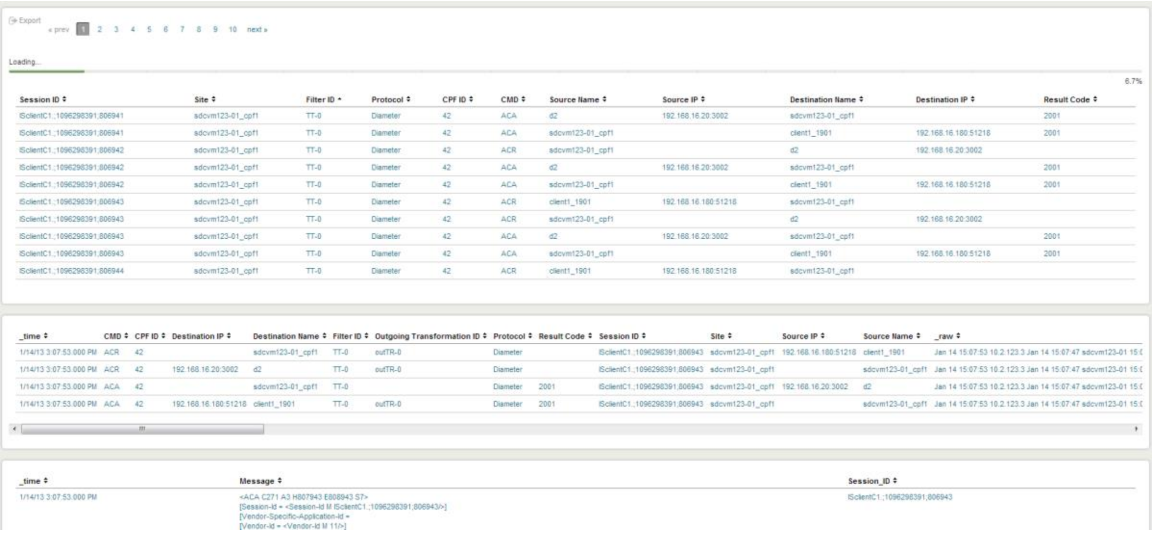
The SDC collects and presents the following properties for each traced message.

Table 64: Traced Message Fields

Traced Field	Description
CPF_ID	The name of the CPF that processed the message.

Traced Field	Description
Filter_ID	The row in the Tracing decision table in the Web UI that the traced message matched.
Protocol	The message protocol.
Command	The message’s command code.
Result Code	The answer that the message returned from the server.
From_Peer_Name	The name of the client peer that the message originated from.
From_Peer_IP	The IP address of the client peer that the message originated from.
To_Peer_Name	The name of message’s destination server peer.
To_Peer_IP	The IP address of the message’s destination server peer.
Peer_Profile	The peer profile of the destination server peer.
Routing_ID	The row in the Routing decision table in the Web UI that the traced message matched.
Session_Binding_ID	The row in the Session Management decision table in the Web UI that the traced message matched.
Incomming_Transformation_ID	The row in the Incoming Transformation decision table in the Web UI that the traced message matched.
Outgoing_Transformation_ID	The row in the Outgoing Transformation decision table in the Web UI that the traced message matched.
Session_ID	The message’s session ID.
Hop_Count	The message’s HOP ID, used internally be the SDC to correlate messages of the same transaction.ss
Message	The content of the message.

Figure 39: Traced Messages – 4 Messages



Export
« prev 1 2 3 4 5 6 7 8 9 10 next »

Loading...

6.7%

Session ID ⇕	Site ⇕	Filter ID ▴	Protocol ⇕	CPF ID ⇕	CMD ⇕	Source Name ⇕	Source IP ⇕	Destination Name ⇕	Destination IP ⇕	Result Code ⇕
ISclientC1.;1096298391;806941	sdcvm123-01_cpf1	TT-0	Diameter	42	ACA	d2	192.168.16.20:3002	sdcvm123-01_cpf1		2001
ISclientC1.;1096298391;806941	sdcvm123-01_cpf1	TT-0	Diameter	42	ACA	sdcvm123-01_cpf1		client1_1901	192.168.16.180:51218	2001
ISclientC1.;1096298391;806942	sdcvm123-01_cpf1	TT-0	Diameter	42	ACR	sdcvm123-01_cpf1		d2	192.168.16.20:3002	
ISclientC1.;1096298391;806942	sdcvm123-01_cpf1	TT-0	Diameter	42	ACA	d2	192.168.16.20:3002	sdcvm123-01_cpf1		2001
ISclientC1.;1096298391;806942	sdcvm123-01_cpf1	TT-0	Diameter	42	ACA	sdcvm123-01_cpf1		client1_1901	192.168.16.180:51218	2001
ISclientC1.;1096298391;806943	sdcvm123-01_cpf1	TT-0	Diameter	42	ACR	client1_1901	192.168.16.180:51218	sdcvm123-01_cpf1		
ISclientC1.;1096298391;806943	sdcvm123-01_cpf1	TT-0	Diameter	42	ACR	sdcvm123-01_cpf1		d2	192.168.16.20:3002	
ISclientC1.;1096298391;806943	sdcvm123-01_cpf1	TT-0	Diameter	42	ACA	d2	192.168.16.20:3002	sdcvm123-01_cpf1		2001
ISclientC1.;1096298391;806943	sdcvm123-01_cpf1	TT-0	Diameter	42	ACA	sdcvm123-01_cpf1		client1_1901	192.168.16.180:51218	2001
ISclientC1.;1096298391;806944	sdcvm123-01_cpf1	TT-0	Diameter	42	ACR	client1_1901	192.168.16.180:51218	sdcvm123-01_cpf1		

_time ⇕	CMD ⇕	CPF ID ⇕	Destination IP ⇕	Destination Name ⇕	Filter ID ⇕	Outgoing Transformation ID ⇕	Protocol ⇕	Result Code ⇕	Session ID ⇕	Site ⇕	Source IP ⇕	Source Name ⇕	_raw ⇕
1/14/13 3:07:53.000 PM	ACR	42		sdcvm123-01_cpf1	TT-0	outTR-0	Diameter		ISclientC1.;1096298391;806943	sdcvm123-01_cpf1	192.168.16.180:51218	client1_1901	Jan 14 15:07:53 10.2.123.3 Jan 14 15:07:47 sdcvm123-01 15:
1/14/13 3:07:53.000 PM	ACR	42	192.168.16.20:3002	d2	TT-0	outTR-0	Diameter		ISclientC1.;1096298391;806943	sdcvm123-01_cpf1		sdcvm123-01_cpf1	Jan 14 15:07:53 10.2.123.3 Jan 14 15:07:47 sdcvm123-01 15:
1/14/13 3:07:53.000 PM	ACA	42		sdcvm123-01_cpf1	TT-0		Diameter	2001	ISclientC1.;1096298391;806943	sdcvm123-01_cpf1	192.168.16.20:3002	d2	Jan 14 15:07:53 10.2.123.3 Jan 14 15:07:47 sdcvm123-01 15:
1/14/13 3:07:53.000 PM	ACA	42	192.168.16.180:51218	client1_1901	TT-0	outTR-0	Diameter	2001	ISclientC1.;1096298391;806943	sdcvm123-01_cpf1		sdcvm123-01_cpf1	Jan 14 15:07:53 10.2.123.3 Jan 14 15:07:47 sdcvm123-01 15:

_time ⇕	Message ⇕	Session_ID ⇕
1/14/13 3:07:53.000 PM	<ACA C271 A3 H807943 E808943 S7> [Session-Id = <Session-Id M ISclientC1.;1096298391;806943/>] [Vendor-Specific-Application-Id = [Vendor-Id = <Vendor-Id M 11/>]	ISclientC1.;1096298391;806943

5.3.9 Previous Release Reports

5.3.9.1 Dashboard

The EMS Dashboard provides a central display of main real-time key performance indicators, statistics graphs and recently generated SNMP traps.

To view the EMS Dashboard:

1. From the tab menu click **Reports > Previous Release Reports > Dashboard**.

Table 46 details and describes the dashboard graphs.

Note: Due to the data processing time, information presented in real-time is presented with a delay of approximately 40 seconds.

Table 65: EMS Dashboard Graphs

Graph	Description
Total Health	The summary of the status of the system resources (snmpd, pacemaker, rsyslogd, traffix_congif_mgr-app, traffic_cpf, etc.). The status of the system resources is queried three times within one minute. The status options are OK (the service/resource is up

Graph	Description
	and working)/Warning (the service/resource was marked as failed at least once in the last minute)/Critical (the service/resource is down)/NA (cannot connect to the service/resource to retrieve the current status). The information is displayed for the last minute, and is refreshed in real-time.
Received/Sent Messages	The total number of received and sent messages by the system. The information is displayed for the last minute, and is refreshed in real-time.
Global Messages per Second	The sum of all incoming and outgoing messages for all CPFs. The information is displayed for the last minute, and is refreshed in real-time.
% Success out of Total Requests	The percentage of successful transactions (answered requests). The information is displayed for the last minute, and is refreshed in real-time.
Global Messages per Second	The sum of all incoming and outgoing messages for all CPFs, by site. The information is displayed for the last minute, and is refreshed in real-time.
Number of Concurrent Sessions	The average number of sessions managed by the SDC Session Repository (Tripo). The information is displayed for the last hour, and is refreshed in real-time.
SNMP Traps	The last 200 traps generated.

5.3.9.2 System View

The System View provides a real-time global view of the system resources.

Note: The EMS Monitoring screens do not display information for hosts and servers in offline SDC sites.

To view the system resources:

Go to **Reports** > **Previous Release Reports** > **System View**. The System View screen is displayed.

1. *Table 67* provides a legend of the different monitoring screen panes.

Table 66: System View

Pane	Description
Site Status	The global number of active and inactive sites (an active site indicates that communication between EMS and the site currently exists, but does not indicate the status of the hosts or services in it)
Host Status	The global number active and inactive of hosts (machines hosting SDC nodes)
Service Status	The summary of the status of the system resources (snmpd, pacemaker, rsyslogd, traffix_config_mgr-app, traffic_cpf, etc.). The status of the system resources is queried three times a minute. The status options are OK (the service/resource is up and working)/Warning (the service/resource was marked as failed at least once in the last minute)/Critical (the service/resource is down)/NA (cannot connect to the service/resource to retrieve the current status).
System Status	Details the sites, hosts, services and resources, their status and its cause.
Site Diagram	Displays the selected site's diagram, detailing the hosts, services and resources (selected in the system status table).

5.3.9.3 System History Status

1. The System History Status provides a real-time global view of the system resources:

To view the system resources:

Go to **Monitoring** > **System History Status**. The System History Status screen is displayed, as shown in *Figure 40*.

To display specific information in the System History Status table:

Using the **Site** and/or **Host** drop-down lists, select a specific site and/or host to display data for.

2.

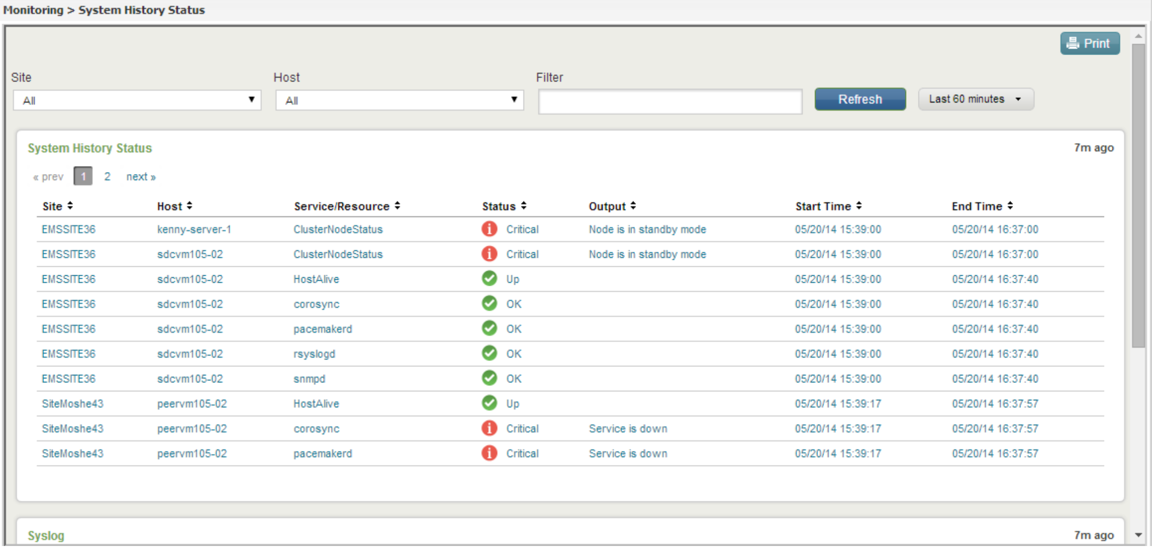
a. Using the **Filter** text box, enter the value that you want to filter the displayed information by.

b. Using the drop-down list next to the **Refresh** button, select the time period that you wish to display data for.

c.

The log messages produced in the selected time resolution will be displayed in the Syslog pane.

Figure 40: System History Status



Site	Host	Service/Resource	Status	Output	Start Time	End Time
EMSSITE36	kenny-server-1	ClusterNodeStatus	Critical	Node is in standby mode	05/20/14 15:39:00	05/20/14 16:37:00
EMSSITE36	sdcvm105-02	ClusterNodeStatus	Critical	Node is in standby mode	05/20/14 15:39:00	05/20/14 16:37:00
EMSSITE36	sdcvm105-02	HostAlive	Up		05/20/14 15:39:00	05/20/14 16:37:40
EMSSITE36	sdcvm105-02	corosync	OK		05/20/14 15:39:00	05/20/14 16:37:40
EMSSITE36	sdcvm105-02	pacemakerd	OK		05/20/14 15:39:00	05/20/14 16:37:40
EMSSITE36	sdcvm105-02	rsyslogd	OK		05/20/14 15:39:00	05/20/14 16:37:40
EMSSITE36	sdcvm105-02	snmpd	OK		05/20/14 15:39:00	05/20/14 16:37:40
SiteMoshe43	peervm105-02	HostAlive	Up		05/20/14 15:39:17	05/20/14 16:37:57
SiteMoshe43	peervm105-02	corosync	Critical	Service is down	05/20/14 15:39:17	05/20/14 16:37:57
SiteMoshe43	peervm105-02	pacemakerd	Critical	Service is down	05/20/14 15:39:17	05/20/14 16:37:57

Table 68 provides a legend of the System History Status table:

Table 67: System Status Table

Column	Description
Site	The name of the site to which the service/resource belongs
Host	The host on which the service/resource runs
Service/Resource	The name of the service/resource

Column	Description
Status	The status of the service/resource
Output	The cause of the service/resource status
Start Time	The date and time that the monitored period began.
End Time	The date and time that the monitored period ended.

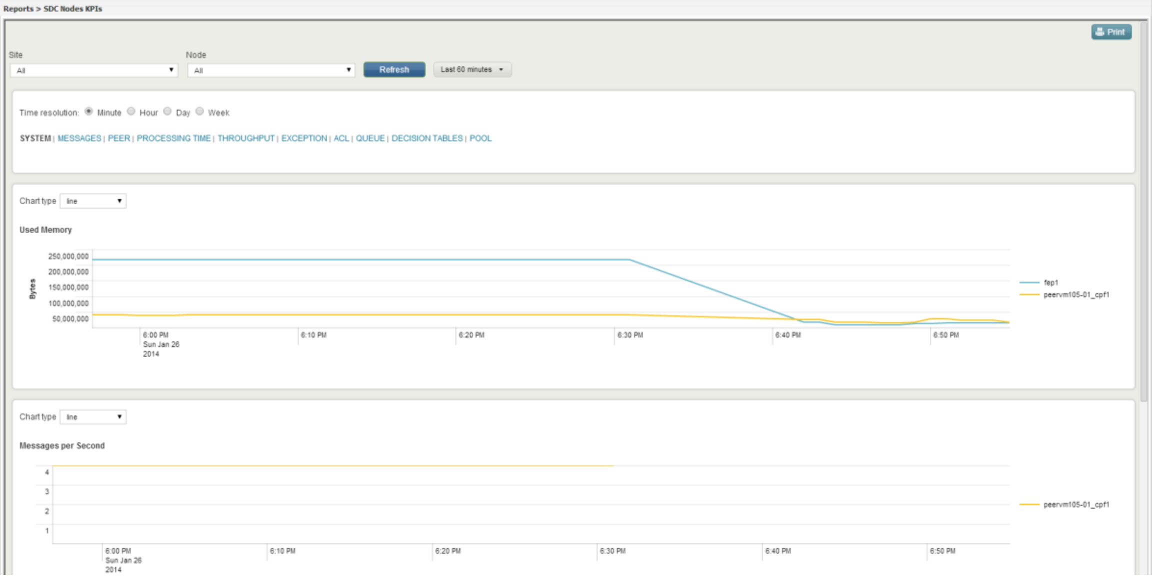
5.3.9.4 SDC Nodes KPIs

The SDC Node KPI reports display SDC node related statistics.

To view an SDC node KPI report:

1. Go to **Reports** > **SDC Node KPIs**. The SDC Node KPIs screen is displayed.

Figure 41: SDC Node KPIs Reports



Reports > SDC Nodes KPIs

Site: All
Node: All
Last 60 minutes
Time resolution: ● Minute ○ Hour ○ Day ○ Week

Chart type: line

2. From the upper part of the screen, select the **Site** and **Node**.

3. Next to the **Refresh** button, select one of the available options in the drop-down menu to define the time period that the data will be displayed for in this screen.

Select one of the **Time resolution** options to define the time resolution in which to display the information in the graphs in this screen. (**Minute**/**Hour**/**Day**/**Week**).

4. Select whether to display graphs related to:

SYSTEM |MESSAGES |PEER |PROCESSING TIME |THROUGHPUT |EXCEPTION |ACL |QUEUE |DECISION TABLES |POOL

5. Select the **Chart Type**.

6. *Table 69* details the available report types:

Table 68: SDC Node KPI Report Types

Category	Report	Description
System	Used Memory	The memory (in MB) that the CPFs and FEPs consumed.
System	Message per Second	The average number of messages processed per second.
Messages	Global Read Limit Bytes Discarded	The number of discarded bytes due to the configured CPFs read rate limit or the rate limit configured per FEP.
Messages	Global Read Limit Message Discards per Second	The number of discarded messages due to the configured CPFs read rate limit or the rate limit configured per FEP or configured per origin peer. The statistic is counted per CPF or FEP.
Messages	Node Read Limit Message Discards per Second	The number of discarded messages due to the configured read rate limit per CPF or per FEP. This statistic is counted per CPF or per FEP.
Messages	Parsed Incoming Messages per Second	The average number per second of incoming Diameter and RADIUS messages (requests and answers) that

Category	Report	Description
		were processed by each CPF per message type.
Messages	Total Parsed Incoming Message per Second	The total number of incoming messages (requests and answers) processed by the CPF or FEP.
Messages	Total Parsed Requests per Second	The total number of requests processed by the CPF or FEP.
Messages	Total Parsed Answers per Second	The total number of answers processed by the CPF or FEP.
Peer	Number of Active Peers	The number of open peers connected to the CPF.
Peer	Number of Peers	The number of peers connected (at present or in the past) to the CPF.
Processing Time	Answer Flow Overall Handle Time	The time period between T3 and T4 of incoming answers, reported by the FEP.
Processing Time	Flow Total Completion Time	The time period between T1 and T4, defined as the total time of a transaction (request and answer).
Processing Time	Answer Flow Handle Time (by Protocol)	The time period between T3 and T4 of incoming answers, per protocol.
Processing Time	Request Flow Handle Time (by Protocol)	The time period between T2 and T1 of incoming requests, per protocol.
Processing Time	Request Flow Overall Handle Time	The time period between T2 and T1 of incoming requests, reported by the FEP-Out
Throughput	Total Processes Received Bytes	The total amount of bytes received and processed by the CPF or FEP.

Category	Report	Description
Exception	Async Executor Rejection Events per Second	The number of requests that are not handled (discarded) due to the CPF overload.
Exception	Message Executor Rejection Events per Second	The number of incoming message events (requests and answers) rejected by the CPF or FEP.
ACL	ACL per Second	The number of client connection requests accepted by the SDC based on the Access Control List.
ACL	Rejected Attempts per Second	The number of client connection requests rejected by the SDC based on the Access Control List.
Queue	Async Task Events Queue Size per Second	The number of requests that are waiting for processing by CPF.
Queue	Incoming Message Events Queue Size per Second	The number of incoming message events (requests and answers) waiting to be handled by the CPF or FEP.
Decision Table	Decision Table per Second	The number of requests handled by a routing/transformation/session management rule.
Pool	Pool 99.95 Percentile of RTT	Pool roundtrip distribution (milliseconds)
Pool	Pool Effective Capacity per Second	The projected pool capacity, based on the combination of the configured rate limit and the real capacity measured in the previous measurement period.
Pool	Pool Health	The pool health percentage (between 0% and 100%), based on the performance in the previous measurement period.

Category	Report	Description
Pool	Percentage of Timeout Events per Second	Percentage of Timeout Events out of total messages counted per pool
Pool	Pool APPLICATION_ERROR Events per Second	Number of APPLICATION_ERROR client pool events
Pool	Pool Overloaded Events per Second	Number of overload events.
Pool	Pool Ramp-Up Overloaded Events per Second	Number of overload events during ramp-up
Pool	Pool TIMEOUT Events per Second	Number of timeout events
Pool	Pool TOO_BUSY Events per Second	Number of too busy events
Pool	Pool Average Roundtrip Time	Pool roundtrip time of messages routed using the pool (milliseconds)
Pool	Pool Sent Messages per Second	Number of sent messages per pool
Pool	Pool Total Answers Received per Second	Number of received messages per pool

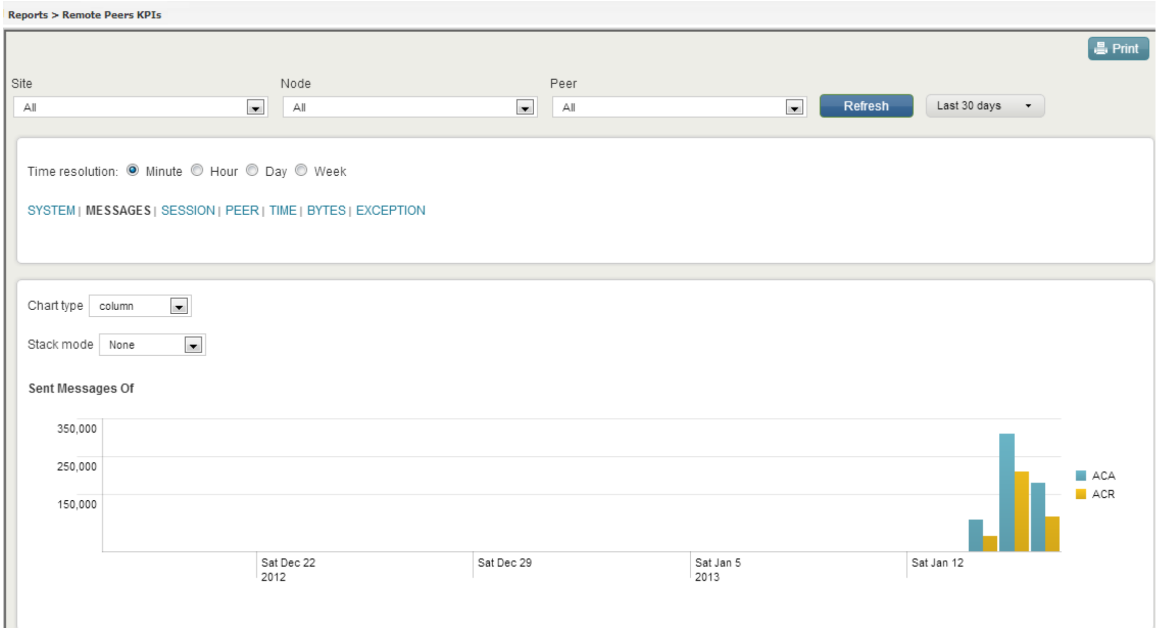
5.3.9.5 Remote Peers KPIs

The Remote Peer KPI reports display the number of sent and failed messages per client per message type per error event.

1. **To view a remote peer KPI report:**

   Go to **Reports > Remote Peer KPIs**. The Remote Peer KPIs screen is displayed.

Figure 42: Remote Peer KPI Reports



Reports > Remote Peers KPIs

Print

Site: All

Node: All

Peer: All

Refresh

Last 30 days

Time resolution: ● Minute ○ Hour ○ Day ○ Week

SYSTEM | MESSAGES | SESSION | PEER | TIME | BYTES | EXCEPTION

Chart type: column

Stack mode: None

Sent Messages Of

2. From the upper part of the screen, select the **Site**, **Node** and **Peer**.

Note: If a **Site**, **Node**, and **Peer** is not selected, graphs will display data for all sites, nodes, and peers.

3. Next to the **Refresh** button, select whether to display data collected in the last 15 minutes, the last 60 minutes, etc.

4. Select one of the **Time resolution** options to define the time resolution in which to display the information in the graphs in this screen. (**Minute**/**Hour**/**Day**/**Week**).

5. Select whether to display graphs related to:

SYSTEM |MESSAGES |PEER |PROCESSING TIME |THROUGHPUT |EXCEPTION

6. Select the **Chart Type**.

7. Select whether to display the reports in **Stack mode** or not.

Table 70 details the available report types:

Table 69: Remote Peer KPI Report Types

Category	Report	Description
System	Discarded Messages (by Message Type) per Second	The number of discarded messages (per message type) due to channel disconnections between the FEP and CPF.
Messages	Peer Local Read Limit Message Discards per Second	The number of discarded messages due to the configured read rate limit per origin peer. This statistic is counted per origin peer.
Messages	Peer Read Limit Message Discards per Second	The number of discarded messages per origin peer. The FEP counter presents the messages that are discarded due to incoming rate limit configuration (per peer and/or per FEP), reported by FEP, and the CPF counter presents the number of discarded messages per FEP. The messages counted are the messages that are discarded due to incoming rate limit configuration (per CPF), reported by CPF.
Messages	Peer Effective Capacity per Second	The projected peer capacity, based on the combination of the configured rate limit and the real capacity measured in the previous measurement period.
Messages	Peer Health	The peer health percentage (between 0% and 100%), based on peer performance in the previous measurement period.
Messages	Sent Messages per Second	The average number of messages sent, counted per destination peer.

Category	Report	Description
Messages	Received messages (by Message Type) per Second	The average number of messages received per second from an origin peer (the total number of received messages in last minute divided by 60 seconds) counted per origin peer per message type. The messages are counted after the incoming rate limit is applied.
Messages	Received Message Before Read Discard per Second	The average number of messages received per second from an origin peer (the total number of received messages in the last minute divided by 60 seconds) counted per origin peer. The messages are counted before the incoming rate limit is applied.
Messages	Sent Message (by Message Type) per Second	The average number of messages routed by the CPF per destination peer. Counted by message type.
Peer	Pending Requests per Second	The average number of requests waiting for an answer per destination peer.
Processing Time	Roundtrip Time	The average time (in milliseconds), of request processing by the destination (T3-T2), counted per source peer and message type.
Processing Time	Peer Average Roundtrip Time	The time period between T2 and T3, defined as the request processing time by destination.
Processing Time	Peer Percentile 99.95% Roundtrip Time	This presents 99.95% of the destination peer latency (T3-T2).
Throughput	Received Bytes	The amount of bytes received, counted per origin peer, before the rate limit.

Category	Report	Description
Throughput	Sent Bytes	The amount of bytes sent, counted per destination peer.
Exception	Retransmission Timeout Events per server per Second	The number of requests that were retransmitted, counted per destination peer and message type. (Counted for RADIUS messages only).
Exception	Timeout Events per Second	The number of unanswered requests due to timeout, per destination peer and per message type.

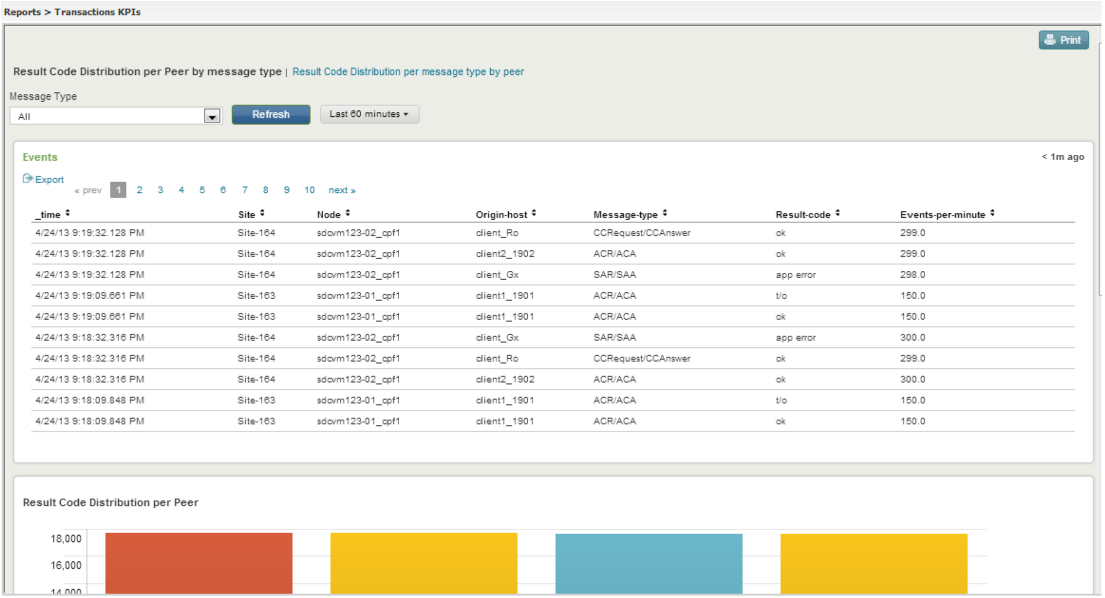
5.3.9.6 Transactions KPIs

The Transactions KPIs reports provide an overview of the SDC’s communication with the server peer – the Remote Node Events that occurred per minute in the selected time frame. This overview can be viewed per server peer (**Result Code Distribution per Peer by message type**) or per message type (**Result Code Distribution per message type by peer**).

To view a Transaction KPI report:

1. Go to **Reports** > **Transactions KPIs**. The Transactions KPIs screen is displayed.

Figure 43: Transaction KPIs



Reports > Transactions KPIs

Result Code Distribution per Peer by message type | Result Code Distribution per message type by peer

Message Type: All

_time	Site	Node	Origin-host	Message-type	Result-code	Events-per-minute
4/24/13 9:19:32.128 PM	Site-164	sdcvm123-02_cpf1	client_Ro	CCRequest/CCAnswer	ok	299.0
4/24/13 9:19:32.128 PM	Site-164	sdcvm123-02_cpf1	client2_1902	ACR/ACA	ok	299.0
4/24/13 9:19:32.128 PM	Site-164	sdcvm123-02_cpf1	client_Gx	SAR/SAA	app error	298.0
4/24/13 9:19:09.661 PM	Site-163	sdcvm123-01_cpf1	client1_1901	ACR/ACA	t/o	150.0
4/24/13 9:19:09.661 PM	Site-163	sdcvm123-01_cpf1	client1_1901	ACR/ACA	ok	150.0
4/24/13 9:18:32.316 PM	Site-164	sdcvm123-02_cpf1	client_Gx	SAR/SAA	app error	300.0
4/24/13 9:18:32.316 PM	Site-164	sdcvm123-02_cpf1	client_Ro	CCRequest/CCAnswer	ok	299.0
4/24/13 9:18:32.316 PM	Site-164	sdcvm123-02_cpf1	client2_1902	ACR/ACA	ok	300.0
4/24/13 9:18:09.848 PM	Site-163	sdcvm123-01_cpf1	client1_1901	ACR/ACA	t/o	150.0
4/24/13 9:18:09.848 PM	Site-163	sdcvm123-01_cpf1	client1_1901	ACR/ACA	ok	150.0

Result Code Distribution per Peer

2. The report displays an event log with the following information: the time stamp, site, node, origin host, message type, result code and events per minute.

   a. If you selected to sort the display by message type, under Message Type, you can filter the information to display a specific message type (for example, CCR/CCA).

   b. If you selected to sort the display by peer, under Peer, you can filter the information to display a specific server peer (for example, PCEF).

The result codes displayed in the Transactions KPIs reports reflect the Remote Node Events that occurred in the selected time frame, as detailed in *Table 71*.

Table 70: Remote Node Event Result Codes

Result Code	Remote Node Events
OK	▪ PeerRemoteNodeEvents_OK
Busy	▪ PeerRemoteNodeEvents_TOO_BUSY
T/O	▪ PeerRemoteNodeEvents_TIMEOUT
App Error	▪ PeerRemoteNodeEvents_CANNOT_ROUTE (deprecated)

Result Code	Remote Node Events
	▪ PeerRemoteNodeEvents_CHANNEL_DISCONNECTED (deprecated) ▪ PeerRemoteNodeEvents_REQUEST_REJECTED ▪ PeerRemoteNodeEvents_REDIRECT ▪ PeerRemoteNodeEvents_APPLICATION_ERROR

5.3.9.7 Session KPIs

The Session KPIs reports display information about session binding and proxy events.

To view a session KPI report:

1. Go to **Reports** > Session KPIs > **Session Statistics**. The Session KPIs screen is displayed.
2. Change the time resolution to which the displayed graphs relate (**Minute/Hour/Day/Week**).
3. Select a report to see the corresponding chart under the report table.

Table 72 details the available report types.

Table 71: Session KPI Report Types

Category	Report	Description
Session Statistics	Proxy On going Session Events Received	The number of session events (updates or terminations) received by the SDC site from its mated SDC site.
Session Statistics	Proxy On going Session Events Sent	The number of session events (updates or terminations) sent by the SDC site to its mated SDC site.
Session Statistics	Successful Bindings Direct Session Events	The number of slave session initiation events that were successfully bound to their defined master session.

Category	Report	Description
Session Statistics	Successfully Handled On-going Direct Session Events	The number of session events (updates or terminations) that were successfully handled by the SDC site.
Session Statistics	Successful Bindings Proxy Session Events	The number of slave session initiation events that were successfully bound to their defined master session by the SDC site and sent to its mated SDC site.
Session Statistics	Successfully Handled On-going Proxy Session Events	The number of session events (updates or terminations) received by a mated SDC site that were successfully handled.
Session Statistics	Un-Successful Bindings Proxy Session Events	The number of slave session initiation events that were received from its mated SDC site and were unsuccessfully bound to their defined master session by the SDC site.
Session Statistics	Un-Successfully Handled On-going Proxy Session Events	The number of session events (updates or terminations) received by a mated SDC site that were not handled successfully.
Session Statistics	Un-Successful bindings Direct Session Events	The number of slave session initiation events that were not successfully bound to their defined master session.
Session Statistics	Un-Successful Handled On-going Direct Session Events	The number of direct (not proxied) session events (updates or terminations) that were not successfully handled by the SDC site.
Session Statistics	Direct Master init success	The number of session initiation events that successfully created master sessions on the SDC site.

Category	**Report**	**Description**
Session Statistics	Proxy Forward Master init success	The number of session initiation events received by the mated SDC site that successfully created master sessions on the mated SDC site.
Session Life Cycle	New Sessions	The number of new sessions.
Session Life Cycle	Session Binding Failures	The number of failed session binding attempts per CPF.
Session Life Cycle	Session Expirations	The number of expired sessions per CPF.
Session Life Cycle	Session Releases	The number of session that were released.
Session Life Cycle	SRR sent on init/terminate sessions	The number of SRRs sent to the mated SDC site for session initiations and session terminations.

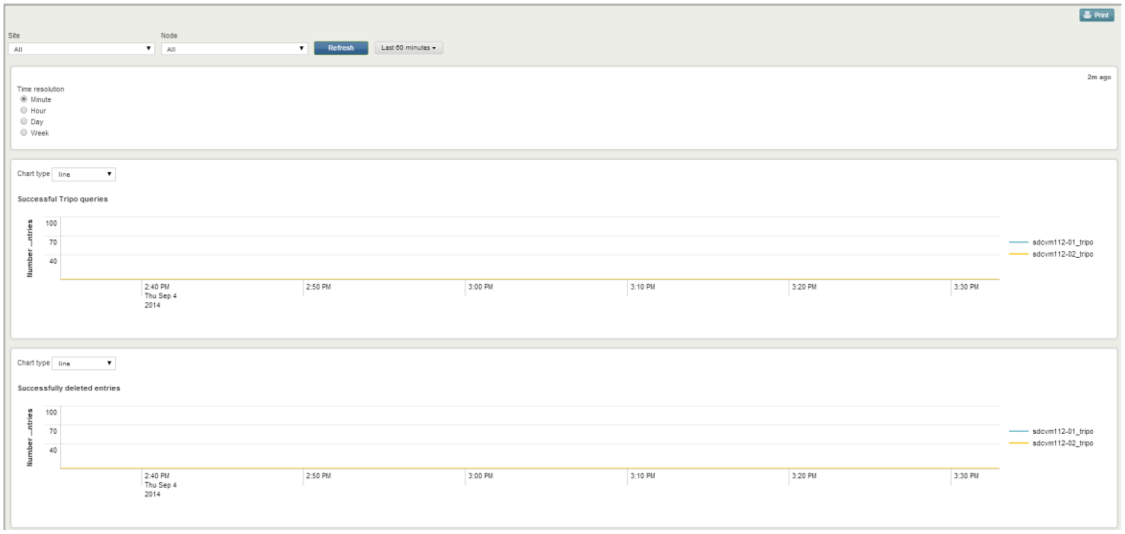
5.3.9.8 Repository KPIs

The Repository KPIs reports display information about sessions saved in the Session Repository.

1. **To view a repository KPI report:**

   Go to **Reports** > **Repository KPIs**. The Repository KPIs screen is displayed.

Figure 44: Repository KPIs Reports



Site: All
Node: All
Last 60 minutes

Time resolution
● Minute
○ Hour
○ Day
○ Week

Chart type: line

Chart type: line

2. From the upper part of the screen, select the **Site** and **Node**.

3. Next to the **Refresh** button, select one of the available options in the drop-down menu to define the time period that the data will be displayed for in this screen.

4. Select one of the **Time resolution** options (**Minute/Hour/Day/Week**) to define the time resolution in which to display the information in the graphs in this screen.

5. Select the **Chart type**.

Table 73 details the available report types.

Table 72: Repository KPI Report Types

Report	Description
Successful Tripo queries	The number of successful Tripo queries per Tripo instance.
Successfully deleted entries	The number of successfully deleted Tripo entries per Tripo instance.
Failed Tripo queries	The number of failed Tripo queries (entry not found) per Tripo instance.
Failed addition attempts (Tripo overflow)	The number of failed additional Tripo attempts as a result of a Tripo storage overflow.

Report	Description
Failed addition attempts (The entry is too long)	The number of failed additional Tripo attempts as a result of the entry being too long.
Failed deletion attempts (entry not found)	The number of failed deletion attempts per Tripo instance (as a result of the entry not being found).
Entry expiration events	The number of Tripo entry expiration events per Tripo instance.
Sent SRRs	The number of acknowledged/failed/expired SRRs sent to the mated SDC site. The statistic is counted per Tripo instance that is sending the SRR.
Sent SRRs during full site replication	The number of acknowledged/failed/expired SRRs sent during full SDC site replication. The statistic is counted per Tripo instance that is sending the SRR.
Sent SRRs during re-synchronization	The number of acknowledged /failed/expired SRRs sent during re-synchronization of the replication queue. The statistic is counted per Tripo instance that is sending the SRR.
Received SRRs	The number of received SRRs successful/failed attempts. The statistic is counted per Tripo instance that is receiving the SRR

5.4 Configuring SNMP Profiles

The SDC supports the following SNMP profiles to retrieve and send information for monitoring purposes:

- SNMPV3 Internal User profile for retrieving OS statistics (such as, CPU, overload per SDC component) to the SDC NMS Agent.
- SNMPv3 and SNMPV2c_Default profiles for external users to retrieve SDC MIB information to an external SNMP application
- SNMP V2- Trap Forwarding for forwarding alarms to an external SNMP application

Note: The SDC is initialized with the default SNMPv2c and SNMPv3 Internal User profiles and only certain parameters are editable. SNMP V2 (including V2c) are supported on IPv6 and IPv4 and SNMPV3 is supported on IPv4.

5.4.1 Retrieving Internal OS Statistics

The SNMPV3_Internal_User profile provides enhanced user security protection to retrieve internal OS statistics per SDC component. This profile is enabled by default and it cannot be disabled.

5.4.1.1 Editing an SNMPv3_Internal User Profile

For SNMPV3_Internal_User profile, you can only edit the Username and Authentication Password and Privacy Password. This is done from the Web UI and by sending the changeSnmpUser API request.

For more information, see the *F5 SDC 5.2 SNMP* Guide.

5.4.2 Retrieving SDC MIB Information to an External SNMP Application

External users can use either the SNMPV2c_Default or SNMPv3 profile to retrieve SDC MIB information (GET PDUs). Dual stack profiles are supported, meaning both SNMPv3 and SNMPv2c_Default external profiles are supported concurrently. While both these profiles provide security settings, they are stronger for SNMPv3 profiles than for the SNMPv2c_Default profile. You can add and edit an SNMPv3 profile, while you can only edit and disable/enable an SNMPv2c_Default profile.

5.4.2.1 Editing an SNMPv2c Default Profile Security Settings

The SNMPv2c_Default profile is used for retrieving SDC MIB information to an external SNMP application. You can edit the default “public” community string to enhance the security setting for this profile.

1.

To edit the Community string for the SNMPv2c_Default profile:

Go to **Administration** > **Specific Site Settings** > **SNMP Settings**.

Select the **SnmpV2c_Default** profile and then select **Edit**.

In the Edit Profile window, in the **Community** field, enter a new security name string.

2.

3. The public community string will no longer work and the new Community security name string will be needed to retrieve SDC MIB information.

Click **Save**.

4. Note: In an EMS deployment, changing the community string must be done from the EMS Web UI.

5.4.2.2 Disabling an SNMPv2c_Default Profile

If you want external users to be able to retrieve SDC MIB information only with an SNMPv3 profile (that has more enhanced security settings), then you need to disable the SNMPv2c_Default profile.

To disable an SNMPv2c_Default profile:

1. Go to **Administration** > **Specific Site Settings** > **SNMP Settings.**

2. Select the **SnmpV2c_Default** profile.

3. Click **Disable** and then **OK** to the prompt question.

Or in the Edit Profile window, unselect the **Enable this profile** checkbox and click **Save**.

5.4.2.3 Adding an SNMPv3 External Profile

An SNMP V3 profile allows external users the ability to retrieve SDC MIB information, with enhanced security protection, to an external SNMP application. To do so, you need to add an SNMP V3 profile and set the security settings.

1.

To add an SNMPV3 profile:

Go to **Administration** > **Specific Site Settings** > **SNMP Settings> Add.**

The Add Profile window is displayed.

2. In the **Profile Name** field, enter a user-friendly name, between 1-32 letters or numbers, without any spaces.

3. In the **Protocol** field, select V3 from the drop-down list.

4. In the **User Name** field, enter a user-friendly name, between 1-32 letters or numbers, without any spaces.

5. In the **Authentication Protocol** field, from the drop-down list, select the authentication protocol (SHA1/SHA2) for security settings for user authentication. The default is SHA1

6. In the **Authentication Password** field, enter a password between 8-32 characters in length, without any spaces.

7. In the **Privacy Protocol** field, from the drop-down list, select the privacy protocol (AES128/ AES192/AES256) for data encryption. The default is AES 128.

8. In the **Privacy Password** field, enter a password enter a password between 8-32 characters in length, without any spaces.

9. Click **Save**.

Once an SNMPv3 profile is configured, you can use it to retrieve SDC MIB information, as in the following Linux command example to retrieve user table information:

```
snmpwalk -v 3 -l authPriv -u userAutomation -a SHA -A authPass -x AES -X privPass 10.240.34.157:1161  .1.3.6.1.6.3.15.1.2.2
```

Note: The default Authentication Protocol, SHA1, is displayed as “SHA -A” and the default Privacy Protocol is displayed as “AES—X”

5.4.2.4 Editing an SNMPv3 Profile

For SNMPv3 profiles, all fields are editable except the **Profile Name** and the **Protocol** (V3).

5.4.2.5 Disabling an SNMPv3 Profile

If you want to remove the enhanced security settings, you can disable an SNMPv3 profile.

To disable an SNMPv3 security settings:

Go to **Administration** > **Specific Site Settings** > **SNMP.**

1. From the SNMP Settings table, select a V3 profile and then click **Disable** and then
2. **OK** to the prompt question.

Or click **Edit** and then in the Edit Profile window, unselect the **Enable this profile** checkbox and click **Save**.

5.4.3 Configuring an SNMP V2-Trap Forwarding Profile

From the SNMP V2-Trap Forwarding profile, you can define the target machines to which SNMP traps are sent.

5.4.3.1 Adding an SNMP V2 Trap Forwarding Profile

1. **To add an SNMP V2-Trap Forwarding profile:**

2. Go to **Administration** > **Specific Site Settings** > **SNMP Settings> Add.**

The Add Profile window is displayed.

3. In, the **Profile Name** field, enter a user-friendly display name, between 1-32 letters
4. or numbers, without any spaces.

5. In the **Protocol** field, select **V2-Trap Forwarding** from the drop-down list.

In the **Community** field, you can change the default Community security name, public, to the security name of the target machine. Enter a name without any spaces.

In the **Host** field, enter the IP address of the target machine.

6. In the **Port** field, enter the port number of the target machine.

7. In the **Timeout** field, enter the time (1-180 seconds), in seconds, that SDC will wait for an answer. The default is 10 seconds.

8. In the **Retry Count** field, enter the defined number of times (1-20) (after a timeout) that the system will try to connect again. The default is 2.

Note: The V2 -Trap Forwarding profile is enabled by default. If you do not want to have the traps forwarded to an external SNMP application, unselect the **Enable this profile** checkbox.

9. Click **Save**.

5.4.3.2 Editing an SNMP V2-Trap Forwarding Profile

For SNMP V2-Trap Forwarding profiles, all fields are editable except the **Profile Name** and the **Protocol** (V2-Trap Forwarding).

5.4.3.3 Disabling an SNMP V2-Trap Forwarding Profile

The SNMPV2 -Trap Forwarding profile is enabled by default. If you do not want to have the traps forwarded to an external SNMP application, disable this profile. You can do this temporarily and then re-enable the profile as needed.

To disable the SNMP V2-Trap Forwarding Profile:

1. Go to **Administration** > **Specific Site Settings** > **SNMP.**
2. Select the **SNMP V2-Trap Forwarding** profile and then select **Edit**
3. Click **Disable** and then OK to the prompt question.

Or in the Edit Profile window, unselect the **Enable this profile** checkbox and click **Save**.

5.5 SNMP Traps

SDC's monitoring and fault analysis abilities are based on SNMP (Simple Network Management Protocol). SDC sends traps to indicate state changes, reaching certain utilization thresholds or encountering unexpected behavior.

To facilitate monitoring and fault analysis in environments where SNMP traps are not supported, SNMP traps are also registered to the log file.

Note: For additional information on log files, see *Logging and Syslog*.

The SNMP community string is set by default to "public".

SDC supports SNMP v2c.

You can also manually configure custom SNMP traps that are included in the relevant MIB files. For more information, see the *F5 SDC SNMP User Guide*.

5.5.1 Stateful Alarms

SDC Stateful Alarms are events that may indicate a performance trend in the SDC. These alarms are automatically raised by the SDC, and remain raised until a "cleared" alarm is generated.

5.5.1.1 Defining Stateful SNMP Settings

The SNMP Settings tab is provided by the SDC to prevent unnecessary stateful alarms from being sent to the defined SNMP targets. By default, all stateful alarms are sent to the defined SNMP targets. For more information about stateful alarms, see *Viewing Stateful*

1. *Alarms.*
2.

To change SNMP Settings properties:

In the **Web UI**, go to **Administration** > **<Specific Site Settings>** > **SNMP**.

Select the **Alarm/Event Settings** tab. The table presents a list of the stateful SNMP alarms and their default SNMP setting. Table 74 details the SNMP settings table properties:

Table 73: SNMP Alarm/Event Settings Table

Column	Description
Event Name	The name of the alarm. e.g., Node State Change
Sent to SNMP Targets	Indicates (true/false) whether the alarm is sent to the defined SNMP targets.

3. Select the relevant alarm, in the **Event Name** column and then click on the respective cell in the **Sent to SNMP targets** column.
4. Select/unselect the checkbox to change the setting from true/false.
5. Click **Submit**.

5.5.1.2 Viewing Stateful Alarms

The stateful alarms are displayed in the **Web UI** (**Alarms>Active Alarms**).

Table 75 describes the table columns, corresponding to the information provided for each alarm. *Table 76* lists and describes the stateful alarms that are generated by the SDC.

Table 74: Active Alarms Table Column Descriptions

Column	Description
Severity	The severity of the raised alarm.
Date and Time	The date and time that the alarm was raised. Note: The displayed alarm time in the Web UI reflects the Web UI browser time, while the alarms are stored in the database based on UTC.
Event Name	The name of the alarm, corresponding to the alarm name in the MIB file.
Message	Information about the circumstances that led to the alarm being raised.
Affected Object Type	The type of SDC component that caused the trap to be raised.
Affect Object	The name of the SDC that caused the trap to be raised.
Host Name	The name of the server running the SDC component that raised the trap.
Detector	The name of the SDC component that raised the trap.
Event Type	The category of events that the alarm belongs to.
Event Id	A unique number assigned to each event type.

Table 75: Available Stateful Alarms

Column	Description of Raised Alarm
FepCpfCommunicationControl	Indicates that control channel between the FEP and CPF is unstable – (the peers' states cannot be exchanged properly and the load balancing mechanism of CPFs may be affected).
GeoSdcProxyNotMarked	Indicates that the virtual server that the peer tried to connect to on the geo-redundant site is not defined as enabled for replication.
GeoSdcProxyConnection	Indicates that the SDC site is not connected to its geo-redundant SDC site.
GeoSdcTripoConnection	The connection between the Session Repository on this SDC and the Session Repository on the geo-redundant SDC site is down. Note: This is an inter-site alarm. In single site deployments, this alarm is generated after an upgrade, even though there is no second site.
GeoSdcTripoFullResyncStarted	Indicates that full session replication between two Tripo instances on two SDC sites has begun.
GeoSdcTripoSrrResyncStarted	Indicates that session replication between two Tripo instances on two SDC sites has begun.
machineDiskPartition	Indicates the current used disk partition, the partition name, the thresholds, the previous used disk partition and the previous alarm severity.
machinePhysicalMemory	Indicates the current used physical memory, the thresholds, the previously measured usage and the previous alarm severity.
peerStateChanged	Indicates the peer's previous state, the new state, the reason for the change and additional information.
peerHealthErrorAnswers	Indicates the percentage of error answers has increased above the configured threshold.

Column	Description of Raised Alarm
poolStateChanged	Indicates the pool’s previous state, the current state, the reason for the change, the number of active peers and whether it is below/above the required minimum.
sdcCmEmsConnection	Indicates that the configuration between the SDC site and the EMS site is not synchronized, since the connection between the configuration manager on the SDC site and the configuration manager on the EMS site is down.
SdcCommuncationofCmCpf	Indicates that the connection between the configuration manager and the CPF is down.
SdcCommuncationofCmFep	Indicates that the connection between the configuration manager and the FEP is down.
SdcCommuncationofCmMateCm	Indicates that the connection between the paired configuration managers is down.
SdcCommuncationofCmNms	Indicates that the connection between the configuration manager and the NMS is down.
SdcCommuncationofCmUi	Indicates that the connection between the configuration manager and the Web UI is down.
SdcCommuncationofCpfTripo	Indicates that the connection between the CPF and the Tripo is down.
SdcCommuncationofFepCpf	Indicates that the message channel between the FEP and the CPF is down, and messages cannot be exchanged.
SdcCommuncationofNmsCpf	Indicates that the connection between the NMS Agent and the CPF is down.
SdcCommuncationofNmsFep	Indicates that the connection between the NMS Agent and the FEP is down.
SdcCommuncationofNmsTripo	Indicates that the connection between the NMS Agent and the Tripo is down.

Column	Description of Raised Alarm
SdcCommuncationofNmsUI	Indicates that the connection between the NMS Agent and the Web UI is down.
SdcComponentFailureRate	Indicates that in the last minute, the percentage of error events measured in the CPF was above the defined threshold.
SdcComponentHealthAnswerQOverload	Indicates that the incoming answer queue is full.
sdcComponentStatus	Indicates if the SDC component is down, up, or status is unknown.
sdcLicenseMPS	Indicates if the volume of traffic processed by the SDC site has exceeded the allowed licensed volume.
vsStateChanged	Indicates the virtual server'sssprevious state, the new state and the reason for the change.

5.5.2 Stateless Alarms

SDC Stateless Alarms are events that do not indicate a performance trend. Rather, they are notifications that report in real-time about a specific event that occurred in the SDC. Stateless alarms do not have multiple severities, rather, they are all issued as a "warning".

5.5.2.1 Defining Stateless Alarm Dilution Settings

The SNMP Dilution Manager is a mechanism provided by the SDC to prevent stateless alarms from flooding the system. Each alarm is assigned a maximum event occurrence number in a specified measuring interval, after which a dilution period, in which no alarms are invoked, begins.

1.

2.

To configure the SDC alarm dilution:

In the **Web UI**, go to **Administration > <Specific Site Settings> > SNMP**.

Click the **SNMP Dilution Manager** tab. The table presents a list of the stateless SNMP alarms and their dilution parameters. *Table 77* details the alarm table properties:

Table 76: SNMP Dilution Manager Table

Column	Description
Event Name	The name of the alarm. e.g., Node State Change
Events in Interval	The number of event occurrences that invoke an alarm, within the specified measuring interval, after which a dilution period begins (during which alarms are not generated). ”0” will stop the alarm generation.
Measuring Interval (Millis)	The interval in which the event occurrences are accumulated, after which a dilution period may begin (during which alarms are not generated).
Dilution Period (Millis)	The period in which no alarms are invoked (begins when the accumulated number of events is exceeded within the measuring interval)

3. Click on a cell in the table to set a new value.
4. Click **Submit**.

5.5.2.2 Viewing Stateless Alarms

Both stateful and stateless alarms are displayed in the **Web UI** (**Alarms**> **Alarm History Log**).

Table 78 describes the table columns, corresponding to the information provided for each alarm. *Table 79* lists and describes the stateless alarms that are generated by the SDC.

Table 77: Alarms Table Column Descriptions

Column	Description
Severity	The severity of the raised alarm.
Date and Time	The date and time that the alarm was raised.
Event Name	The name of the alarm, corresponding to the alarm name in the MIB file.
Message	Information about the circumstances that led to the alarm being raised.
Affected Object Type	The type of SDC component that caused the trap to be raised.
Affect Object	The name of the SDC that caused the trap to be raised.
Host Name	The name of the server running the SDC component that raised the trap.

Column	Description
Detector	The name of the SDC component that raised the trap.
Event Type	The category of events that the alarm belongs to.
Event Id	A unique number assigned to each event type.

Table 78: Available Stateless Alarms

Column	Description
SdcComponentGcLoop	Indicates that a Garbage Collector loop was detected. The old generation heap size after GC is above the defined threshold.
SdcCustomAlarm	Indicates an alarm that was created manually by the user.
SdcDnsResolvingSuccess	Indicates that the DNS resolving of a given destination succeeded.
SdcFileServerCloseFile	Indicates that a degraded file closing attempt failed.
SdcFileServerDirectory	Indicates that a new directory creation attempt has failed.
SdcFileServerFileCreate	Indicates that a new degraded file creation attempt failed.
SdcFileServerRenameFile	Indicates that a degraded file renaming attempt has failed.
SdcFileServerSplitFile	Indicates that a file split attempt has failed.
SdcMaxTracePerDayReached	Indicates that the number of traced transactions reached the maximum allowed number of traced transactions per day.
SdcMaxTraceTPSReached	Indicates that the number of traced TPS reached the maximum allowed volume of traced TPS per day.
SdcPeerAclRejected	Indicates that there is no rule in the Access Control List allowing this peer connection.
SdcPeerCapacityReached	Indicates that the SDC site is already configured with the maximum allowed number of peers.
SdcProcessRestart	Indicates that an SDC process was restarted.
SdcScriptInvocationFailed	Indicates that the script failed.
SdcUserAuthenticationFailure	Indicates that the user credentials entered are not authorized to access the system.

5.5.3 SNMP Logs

To facilitate monitoring and fault analysis in environments where SNMP traps are not supported SNMP traps are logged to SDC’s log files.

Log messages appear in the following format: **SNMP** Alarm was created: <NOTIFCATION TEXT>, with properties: <ALL TRAP PROPERTIES> **SNMP**

5.6 Logging and Syslog

The SDC events are logged according to their nature (e.g.: system, networking, etc.). Log messages (FEP, CPF, and NMS Agent) are stored in the local file system of each node and can be configured to be sent from a locally installed Syslog client to a remote Syslog Daemon. The Syslog Daemon and log detail level of each event that triggers log recordings are configured in the SDC Web UI.

Note: When log messages exceed 1028 bytes minus the timestamp and header, the text is displayed as truncated. When you see "..." at the end of a log message, check the following message for the remaining text.

5.6.1 Setting the Log Levels

You can set the log level depending on the detail level that you want to log.

1. **To set the SDC log levels:**
2. Go to **Administration** > **Specific Site Settings** > **Logging**.

From the **Log Levels** tab select a log level from the drop-down list for each category. For example, for **Configuration**, select an **INFO** log level and for **Networking** a **WARN** log level.

Note: SDC prints all logs of the selected log level and also those of above log levels.

Table 80 describes the different Log Detail Levels.

Table 79: Log Detail Level

Level	Description
Fatal	Indicates very severe error events that presumably lead to application abort, such as: unexpected shutdown, component init/start failure, configuration load failure, and memory exhaustion, virtual server binding or listening failure.
Error	Indicates negative oriented events that might still allow the application to continue running, Error log message may indicate major traffic damage due to server/flow manager malfunction or queue overload. Such event may be: abnormal peer disconnection, peer connection attempt failure, script loading failure, major fitness degradation of Server Remote Nodes or SDC itself.
Warn	Indicates potentially harmful situations, pointing out a certain threshold is exceeded in a predefined time interval. Such event may be: the number of message (transaction) errors, script runtime exceptions, routing failures, parsing failures, message creation failures.
Notice	Indicates positive oriented events that point out the progress of the application at a coarse-grained information level. Such events may be: normal peer disconnection, successful peer connection, component startup info, configuration changes, system status summary, statistics summary, flow manager failures, fitness level improvement (of Server Remote Nodes or SDC itself).
Info	Indicates message related events that highlight the progress of the application at a coarse-grained information level. Such events may be: transaction completion state, incoming request or answer, outgoing request or answer and failure conditions such as timeouts, error in answer, missing pending request.
Debug	Indicates events that are most useful to debug an application with, at a fine-grained information level. Debug log level is similar to Info log level, only it holds message content.

Level	Description
Trace	Indicates events that are most useful to debug an application with, at a finer-grained information level than the Debug level.

Table 80: Customized Log Level Categories

Log Category	Description
Administration	Reports of events related to system administration such as changes made to the system configuration, including identity of the administrator.
Peer	Reports of events related to Remote Peers
Protocol	Reports event related to the network protocol
Transaction Management	Reports of events related to transaction flow through the system
Storage	Reports events related to User Data Storage
System	Reports of events related to the system such as resource failures (no memory, file not found, disk full, etc.), unknown exceptions, system initializations and terminations.
Networking	Reports of events related to networking
Configuration	Reports of events related to system configuration such as peer configuration, routing table, etc.
SNMP	Reports of event that trigger SNMP
User Trace	**Reports of events that are user traced (specifically traced by the user via scripts)**

3. Note: The log level of a category cannot extend the log level as defined in log4j.xml.

Click **Submit** to save the log settings.

5.6.2 Enabling the Session Life Cycle and Session Error Logs

The SDC can be configured to create logs for session life cycle events and session errors. These logs can be used to help troubleshoot when stateful sessions fail to route.

The location of the logs for regular and Session Repository errors, respectively, is under:

/opt/traffix/sdc<build>/logs/<cpf instance name>_cpf1/session_output

/opt/traffix/ sdc<build>/logs/<cpf instance name>_cpf1/session_error

When enabled, the following events/errors are written to the log files with the information shown in *Table 82*.

Table 81: Life Cycle Events Written to the Session Output Log File•

Event	Related information written to the log file
Session created on a local CPF by a local peer or by an SRR message	▪ Time Stamp ▪ Session ID ▪ Session Action (Created sessions are indicated with a "C" tag) ▪ Origin Peer ▪ Destination Pool ▪ Destination Peer ▪ Session Type: M=Master /S=Slave/LU=Session Lookup Note*:* A persistent session that has no binding key will appear as a master session. ▪ Master Session ID Note: This is displayed for slave sessions only. ▪ SM Row ID ▪ Binding Keys Note: This is displayed for master sessions only. ▪ Published keys (“P”) are for persisted session data ▪ Lookup Keys are used to lookup persisted session data

Event	Related information written to the log file
	▪ Session Sources (Local creation indicated with an "L" tag or by SRR message indicated with an "SRR" tag) ▪ Timeout
Session removed from local CPF due to expiration	▪ Time Stamp ▪ Session ID ▪ Session Action (Removed sessions are indicated with an "R" tag) ▪ Session Release (Expired sessions indicated with an "EX" tag) ▪ Session Sources (Local creation indicated with an "L" tag)
Session removed from local CPF due to session release	▪ Time Stamp ▪ Session ID ▪ Session Action (Removed sessions are indicated with an "R" tag) ▪ Origin Peer ▪ Destination Pool ▪ Destination Peer ▪ Session Binding Row-ID ▪ Session Release (Released sessions indicated with an "RE" tag) ▪ Session Sources (Local creation indicated with an "L" tag or by SRR message indicated with an "SRR" tag)
Session removed from local CPF based on SRR message	▪ Time Stamp ▪ Session ID ▪ Session Action (Removed sessions are indicated with an "R" tag) ▪ Origin Peer ▪ Session Release (Released sessions indicated with an "RE" tag) ▪ Session Sources (By SRR message indicated with an "SRR" tag)
Error events	▪ Time Stamp ▪ Session ID ▪ Reason for failures ▪ TD – Tripo is down

Event	Related information written to the log file
	- SD – replication site is down - NF – a session is neither found in a repository nor found in a session management table - IK – null binding key found in a slave session - IL – no valid key found for a session lookup - BF – binding failure - LF – no matching session found for a lookup session - Origin Peer - Session Sources ("SRR" tag) Note: This information is only logged for SRR errors. - Tripo Action that failed Note: This information is only logged for TD (Tripo down) errors.

To enable session logging:

1. Go to **Administration** > **Specific Site Settings** > **Logging**.
2. Select the **Enable Session Log** checkbox**.**

1. **To add session attributes to a session log:**

Under **Entity Attribute**, type in the attribute you want to add to the session log message.

Note: Use the following syntax: <Element>.<Property> and not the syntax from the groovy method that includes "( )." For more information, see *Appendix D: Decision Table Attributes*Appendix D:

2. Under **Description**, type in the description for the attribute.

The added attributes to the session logs are generated at the end of the standard log message, and are delimited by “%”. For example:

Session.slave.xxx;C;origin_host_1;pool1;s_4000;S;Session.master.xxx;SB-2[host: host_name];L;271%12345%

3. Click **Submit**.

Note: All logging is done in batches, i.e. accumulating 16K of log data before writing it to the log file. This means that there might be logging data which is still in memory. Any engineering script can be invoked (**Administration** > **Engineering Scripts** > **NMS Engineering Scripts** > **Print Status to Log)** to flush the remaining log data to the log file.

5.6.3 Defining Syslog Daemon Addresses

You need to define the IP addresses so that log messages can be automatically sent from a locally installed Syslog client to a remote Syslog Daemon.

Note: Only when using the direct forwarded method, CPF and FEP log messages that are sent to a remote Syslog Daemon can be configured in the Web UI. For logs, including ELK application logs, that are centrally forwarded, from the OAM, the configuration is done with a Salt API as described in the *F5 SDC Bare Metal System Maintenance Guide*.

To add an SDC Syslog Addresses:

1. Go to **Administration** > **Specific Site Settings** > **Logging** > **Syslog Addresses**.

Table 83 presents a list of Syslog Daemon Addresses properties.

Table 82: Syslog Addresses

Column	Description
IP Address:port	The IP address to which log files are sent.
Facility	Indicates the software type (auth, authpriv,daemon, cron, ftp, lpr, kern, mail, news, syslog, user, uucp, or local0 ... local7) that generated the message.

2. Click **Add,** and then define its **IP address** and **Facility.**

Repeat this step for any additional Syslog Daemons that should receive the log message output.

Click **Submit** to save the log settings.

3.

5.6.4 Log File Size Control

4. Log messages are stored in the local file system of each node and can be sent to a remote server via syslog. Each node's log file size control is configured with a maximum threshold. The threshold parameters are configured in the log4j.xml file and *Table 84* shows their default values.

Table 83: Log File Size

Parameter	**Default Value**
MaxBackupIndex	10
MaxFileSize	10MB

5.7 Tracing

SDC provides you with the ability to capture all signaling traffic passing through the system and examine specific signaling flows in all the supported protocols. The transmitted data is captured when a transaction's AVPs match a tracing rule. The transaction's requests and answers are then logged and can be viewed in the **Reports** tab (for additional information, see Reports).

5.7.1 Configuring a Tracing Rule

Before configuring a tracing rule to capture transaction data, you need to define the relevant tracing attributes.

5.7.2 Defining Tracing Rule Attributes

This section describes how to configure a tracing rule attribute.

To add an Association Rule attribute:

1. Go to **Administration** > **Tracing** > **Rule Attributes**. The Rule Attributes window displays the list of attributes that may be used to define the tracing rules:
2. Click **Add**. A new line is added to the table.
3. Under **Label**, type in a user friendly name that will be used to identify the attribute. e.g.: “OriginHost”.
4. Under **Attribute**, type in the name of the AVP retrieved from the message. e.g.: “request.Origin-Host”
5. Under **Type**, select the data type of the new attribute. e.g.: String
6. Under **Description**, type in a short description of the attribute.
7. Click **Save**.

Note: For additional information on the decision table attributes, see *Appendix D: Decision Table Attributes*.

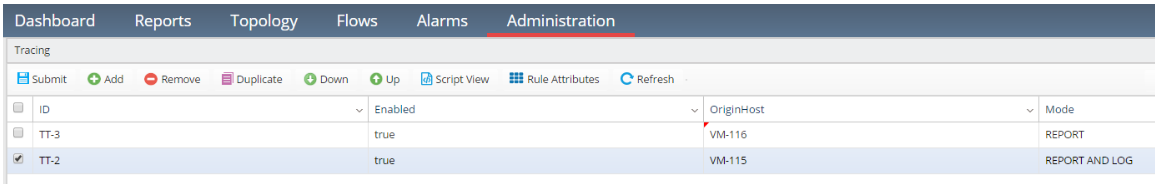
5.7.3 Adding a Tracing Rule

The Tracing table’s columns represent the previously defined Tracing Rule Attributes. If you have not set any attributes, see *Defining Tracing Rule Attributes*.

To add a new tracing rule:

1. Click **Add** to create a new tracing rule. A new rule line assigned an automatic name is added to the table.

Figure 45: Tracing Rules



☐	ID	Enabled	OriginHost	Mode
☐	TT-3	true	VM-116	REPORT
☑	TT-2	true	VM-115	REPORT AND LOG

2. Under each column, select the value against which messages are compared. For example: under **OriginHost** set the value “VM-l15”. This rule shall apply to messages originating in VM-l15 and these messages’ data will be captured.

3. Under **Mode**, select how to display the traced data according to the following drop-down options:

- **REPORT –** the traced data is sent to the EMS site, and is not written to the local log file.
- **REPORT AND LOG** – the traced data is sent to the EMS site, and is also written to the local log file.
- **REPORT AND LOG WITH HEX-DUMP** – the traced data is sent to the EMS site, and is also written to the local log file with a hex dump.
- **LOG TO LOCAL –** the traced data is only written to the local log file.
- **LOG TO LOCAL WITH HEX-DUMP –** the traced data is only written to the local log file with a hex dump.

Note: The number of traced messages is limited to 1000 TPS per site. The maximum traced bytes per site per day is 10 GB.

Deleting a row in the Tracing table will not stop the tracing requests of a persisted session, as they are traced until the session is expired. You must disable the relevant tracing rule in the Tracing table if you want to stop tracing the requests of a persisted session

6. Managing the SDC

This chapter describes how you can manage the SDC configurations.

6.1 Restoring Previous Configurations

SDC Web UI provides its users with a simple basic set of rollback actions. In case SDC is not operating as expected and the cause of the unexpected behavior is unknown, a previous configuration setting can always be restored and used. The user may choose to restore a configuration set assembled when a specific audited action was performed, or to restore a setting of an initiated backup snapshot.

The auditing feature captures the configuration actions taken by the system's users. Users may add a Remote Node, modify a Transformation script, edit a Pool or perform any configuration change, depending on their privileges. All actions are documented. Each user action is saved to a separate entry. Each entry is registered with a time stamp, the performing user and the type of performed action.

In addition to the restore option available from the audited actions list, you may easily initiate a backup of the SDC's current configuration, creating a safe snapshot of the configuration and restore that configuration at any given moment.

6.1.1 Auditing

Each of the UI actions taken by the SDC's users is documented and registered to the auditing list. You may select any of the audited actions to restore the documented configuration of the exact point in time that the action was performed.

The following actions are examples of the audited actions:

- Adding a Remote Node
- Adding a Health Monitor
- Add a Pool
- Backup

- Changing a Flow script
- Changing a Health Monitor
- Changing a Routing Script
- Change a Transformation Script
- Change a User Tracing script
- Changing a Cluster Node’s configuration
- Editing a Pool
- Editing a Remote Node
- Removing a Remote Node
- Removing a Pool
- Deleting a Script
- Renaming a Script
- Restoring a previous configuration
- Removing a data dictionary
- Setting the SNMP dilution.
- Setting a log level
- Setting a log level and the syslog address
- Changing the onSessionCreate and onSessionRelease scripts
- Changing the onCollectPerformanceRecords script

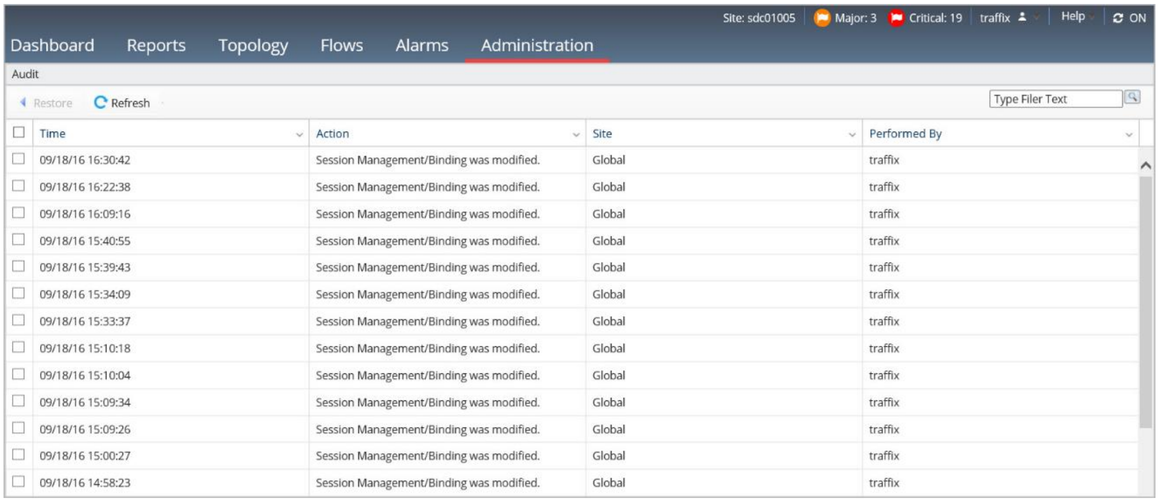
1.

To view the audited entries:

Go to **Administration** > **Audit.** The Audit screen is displayed.

Note: The maximum number of audit entries that are displayed is 200.

Figure 46: Audit



□	Time	Action	Site	Performed By
□	09/18/16 16:30:42	Session Management/Binding was modified.	Global	traffix
□	09/18/16 16:22:38	Session Management/Binding was modified.	Global	traffix
□	09/18/16 16:09:16	Session Management/Binding was modified.	Global	traffix
□	09/18/16 15:40:55	Session Management/Binding was modified.	Global	traffix
□	09/18/16 15:39:43	Session Management/Binding was modified.	Global	traffix
□	09/18/16 15:34:09	Session Management/Binding was modified.	Global	traffix
□	09/18/16 15:33:37	Session Management/Binding was modified.	Global	traffix
□	09/18/16 15:10:18	Session Management/Binding was modified.	Global	traffix
□	09/18/16 15:10:04	Session Management/Binding was modified.	Global	traffix
□	09/18/16 15:09:34	Session Management/Binding was modified.	Global	traffix
□	09/18/16 15:09:26	Session Management/Binding was modified.	Global	traffix
□	09/18/16 15:00:27	Session Management/Binding was modified.	Global	traffix
□	09/18/16 14:58:23	Session Management/Binding was modified.	Global	traffix

Table 85 presents a list of audited actions taken by SDC users.

Table 84: Audit Entries Properties

Column	Description
Time	The date and time on which the configuration change occurred.
Action	The configuration change.
Site	The site to which the configuration change was applies (or “Global” (EMS) if the configuration change was applied to all sites).
Performed By	The user that performed the configuration change.

1.

To refresh the Audit table:

1. Click **Refresh**.

2. **To restore a previous configuration mode:**

Select the Audit entry you want to rollback.

Click **Restore**.

Note: In the EMS Web UI, all UI actions performed in both the EMS site and the SDC sites managed by the EMS site are displayed in the Audit list. You can roll back an action performed on an SDC site using either an SDC or EMS Audit Web UI. You cannot roll back an action performed on an EMS site, using an SDC Audit Web UI. When performing an audit, after an EMS upgrade, you must restart the remote sites so that all audit data relating to the remote sites is displayed in the Audit screen.

Warning: Selecting to rollback a specific audited action will roll back every audited action performed subsequently to the selected change (i.e.: every entry above the selected entry will rollback too).

6.1.2 Backup & Restore

The user may easily initiate a backup of the SDC's current configuration, creating a safe snapshot of the configuration and restore that configuration at any given moment.

To view the list of backup snapshots:

1. Go to **Administration** > **Backup And Restore**. The Backup And Restore screen is displayed.

Table 86 presents a list of backup snapshots actions taken by the SDC users.

Table 85: Backup Snapshot Properties

Column	Description
Time	The date and time on which the backup was performed.
Snapshot	The name of the backup snapshot, given by the performing user.
Performed By	The user that performed the backup.

1.

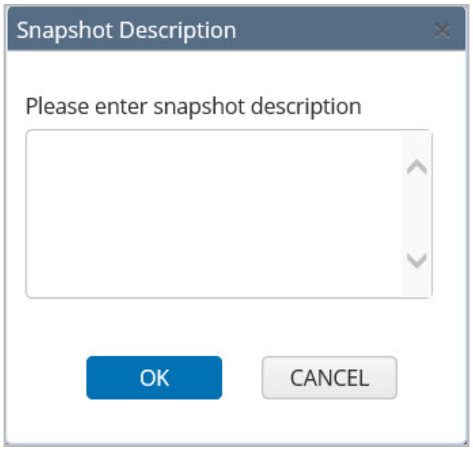
To refresh the Backup And Restore table:

1. Click **Refresh**.

To backup the current configuration and create a snapshot of SDC:

Click **Backup**. The Snapshot Description dialog box is displayed.

Figure 47: Snapshot Description



Snapshot Description

Please enter snapshot description

Enter a meaningful description for the current SDC configuration.

2. Click **OK**. The new backup snapshot appears in the Backup Snapshots table.

3.

To restore a backup snapshot:

1. From the backup snapshots table, select the snapshot you want to restore.

2. Click **Restore**.

6.2 User Management

Note: If you are using a third-party LDAP authentication system, this Web UI section will be disabled.

To keep a secure system, SDC maintains an effective user management system, allowing privilege hierarchy through simple and effective user account management techniques. Each user is given a unique identity and a predefined set of privileges with which SDC may be configured.

The user management mechanism authenticates users according to usernames and passwords. SDC administrators can configure a password policy by setting password requirements.

Users are authorized according to their given roles, which can be configured with expiration dates. SDC administrators can add new users, remove existing users, or edit the roles or expiration dates of existing ones. In addition, SDC administrators can temporarily not allow a user access to the system, by "disabling" a user. In the event that a user is locked out, because the user exceeded the number of allowed login attempts within a certain time period, an SDC administrator can unlock the user's account.

Note: The functionality of administrators disabling user access is limited to user type “user.” Other user types cannot be locked out.

Table 87 details the user roles and their privileges:

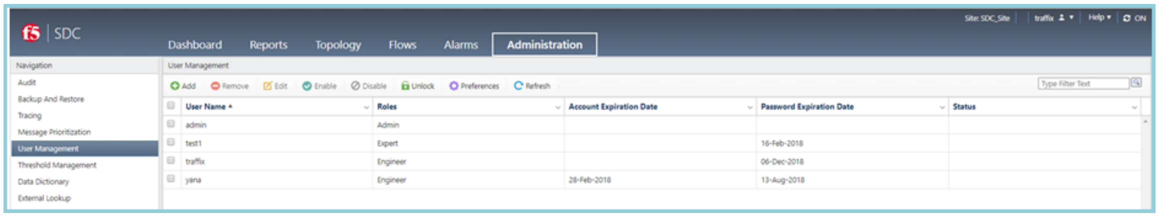
Table 86: User Type Privileges

User Type	Privileges
Engineer	Write engineering scripts, view engineering statistics.
Admin	Perform Configuration changes, submit them and create new users via User Management.
Expert	Perform configuration changes and submit them.
User	View the configuration without performing any changes.

1. **To create a new user in SDC Web UI:**

   Go to **Administration** > **User Management**. The User Management screen is displayed.

Figure 48: User Management



User Name	Roles	Account Expiration Date	Password Expiration Date	Status
admin	Admin			
test1	Expert		16-Feb-2018	
traffix	Engineer		06-Dec-2018	
yana	Engineer	28-Feb-2018	13-Aug-2018	

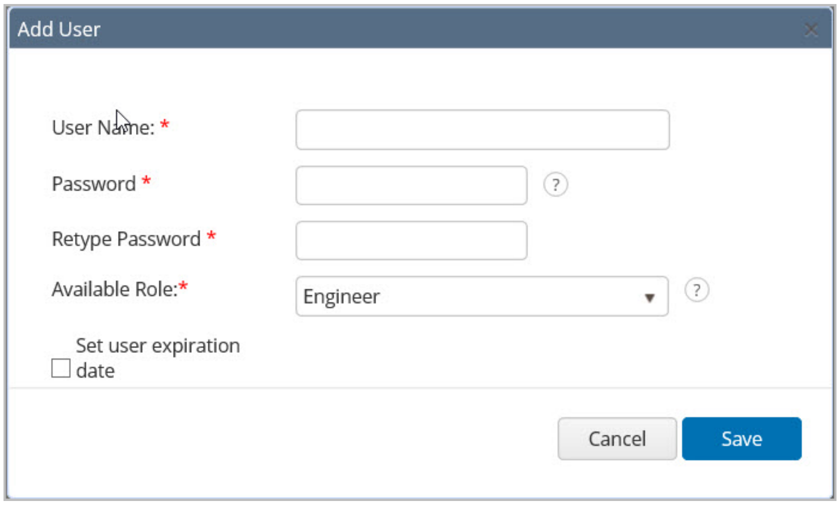
Table 88 presents a list of the SDC users.

Table 87: SDC Users

Column	Description
User Name	The user’s unique identifier.
Roles	The list of privileges the user is assigned with.
Account Expiration Date	The date up to in which the user is authorized
Password Expiration Date	The date up to in which the password is in effect
Status	User is enabled/disabled

Click **Add**. The Add User dialog box appears

2. **Figure 49: Add User**



Add User

User Name: *

Password *

Retype Password *

Available Role:* Engineer

□ Set user expiration date

3.

4.

In **User Name** field, enter the user’s unique identifier.

5. In **Password** field, enter the user’s password and retype it in **Retype Password**
6. field.

From the **Available Role** drop-down, select the role you want to assign to the user.

7.

8. Select **Set user expiration date** and fill in the relevant date if you want the selected user to only have the privileges for a certain time period.

Click **Save**. The role is added to user’s role list.

Repeat the above steps for each role you want to add to the list.

Note: All roles below the selected user level are automatically assigned to the user.

To remove any user from the list:

Select the row of the user you want to remove.

Click **Remove**.

1.

To refresh the user list:

2.

Click **Refresh**.

1. **To edit a user's password or expiration date:**

1. Select the user from the **User Name** list and click **Edit**. The Edit User dialog box appears.

To enable/disable a user from accessing the system:

1. Select the row of the user you want to enable/disable.

2. Click **Enable/Disable**.

1. **To unlock a user's account:**

2. Select the row of the user (who has been locked out of the system) that you want to unlock.

Click **Unlock**.

6.2.1 Configuring a Password Policy

SDC administrators (and engineers) can configure a password policy by setting password requirements and a lockout policy. The policy sets the required length and complexity of the password. In addition, SDC administrators (and engineers) can set password expiration periods and account lockout policies (that sets how many login attempts are allowed within a configured time period).

To configure the password policy:

1. Go to **Administration** > **User Management** > **Preferences**.

   The Preferences window opens with the default settings.

2. Configure any of the following fields:

   a. **Minimum password length**

   b. **Maximum password length**

   c. **Password expiration (days)**

   d. **Warning expiration (days)**

   e. **Enforce history**

   Note: This sets the number of unique new passwords that have to be associated with a user account before an old password can be reused.

   f. **Minimum lower case** (a-z)

   g. **Minimum upper case** (A-Z)

   h. **Minimum digits** (1,2..)

   i. **Account lockout threshold –** number of attempts

   j. **Account lockout time –** period in minutes for login attempts

   k. **Auto unlock account–** enables the auto unlock and defines the lockout period, after which the user will be able to login again

3. Click **Save**.

6.3 FTP servers

FTP Servers are used to retrieve information saved to the file server in the Offline Processing Mode. For more information, see *Appendix C: Offline Processing Mode*.

6.4 Applying Engineering Scripts

Engineering scripts can be used to troubleshoot the status (connectivity) of SDC components. Only users who are defined as engineers can apply engineering scripts to the following components:

- Cluster Nodes– to CPF, FEP and configuration manager components (SDC sites only)
- Web UI

NMSWarning: Engineering scripts should only be applied in consultation with F5 Support

To apply an engineering script to an SDC site:

1. Go to **Administration > Engineering Scripts**.

a. To apply to a FEP, CPF, configuration manager component, select the following:

i. **Cluster Nodes Engineering Script.**

ii. In the **Name** field, enter the SDC component name as it appears in the SDC Component Name column.

b.

c. To apply to a Web UI component, select **UI Engineering Script**.

2. To apply to a NMS Agent component, select **NMS Engineering Script**.

1. Click **Submit**.

To apply an engineering script to an EMS site:

a.

b. Go to **Administration > Engineering Scripts**.

2. To apply to a Web UI component, select **UI Engineering Script**.

To apply to an NMS Agent component, select **NMS Engineering Script**.

Click **Submit**.

Appendix A: User Data Storage

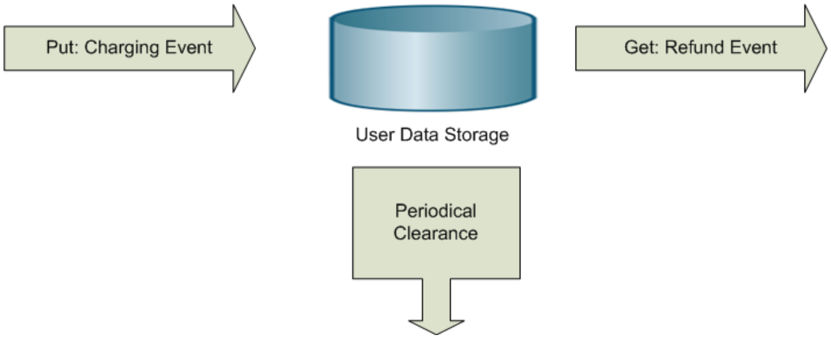
SDC allocates a special memory hook on which you may create and maintain simple and complex data structures. The memory hook is called User Data Storage. The User Data Storage is typically used to store cross-session data (e.g. client details).

The data structures in the User Data Storage may be used to store data in and draw data from, when needed. They are created and maintained via SDC's Flows and administration Groovy scripts.

There are two types of User Data Storage – Persistent and Transient. The transient User Data Storage is local to SDC and is kept within the SDC memory: it exists as long as SDC is ON and destroyed when SDC shuts down. The persistent storage is duplicated for persistency and Redundancy. The type of User Data Storage is configured throughout the SDC installation procedure. The selected type is referred to as the default type.

Since both data storage types are session-independent, the SDC user is responsible for their periodical clearance. The storage clearance interval should be set according to the data usage. For example: if, according to company's policy, the information may be accessible within the 24 hours following a business transaction, the user storage should be cleared once every 24 hours. The clearance interval also limits the volume of the data that can be stored.

Figure 50: User Data Storage



The User Data Storage may be arranged in any data structure, the choice is up to the user's decision is expressed in the Groovy scripts that access the User Data Storage:

- Array
- Matrix
- Tree
- Etc.

Traditionally, the way to manage the User Data Storage is:

1. Getting an instance of the storage provider factory:

2. ▪ public static StorageProviderFactory getInstance();

Creating a table:

▪ public <K, V> StorageProvider<K, V> createUserTable(String tableName);

3. ▪ public <K, V> StorageProvider<K, V> createUserTable(String tableName, long

4. lifespan);

Performing table operations (see the following implementation example)

Optionally retrieving a table:

▪ public <K, V> StorageProvider<K, V> getUserTable(String tableName).

A.1 Implementation Example

The following script is an example of performing table operations.

```
userTraceLogger.debug("test external storage started");
            def factory = StorageProviderFactory.getInstance();
            def id = System.currentTimeMillis();

            def keyList = new ArrayList();
            keyList.add("k1-" + id);
            keyList.add("k2-" + id);
            keyList.add("k3-" + id);

            userTraceLogger.debug( ("createUserTable");
            def createdTable = factory.createUserTable("myTable");

            userTraceLogger.debug("putNow/putAllNow");
            createdTable.putNow("test1-a-" + id, "t-1-a");
            createdTable.putNow("test1-b-" + id, "t-1-b");
            createdTable.putAllNow(keyList, "t-1-mult");

            userTraceLogger.debug("getUserTable");
            def table = factory.getUserTable("myTable");
            assert table.get("test1-a-" + id).equals("t-1-a") : "expected to find:
t-1-a but found: " +
            table.get("test1-a-" + id);
            assert table.get("test1-b-" + id).equals("t-1-b") : "expected to find:
t-1-b but found: " +
            table.get("test1-a-" + id);
            assert table.get("k1-" + id).equals("t-1-mult") : "expected to find: t-
1-mult but found: " + table.get("k1-"
            + id);
            assert table.get("k2-" + id).equals("t-1-mult") : "expected to find: t-
1-mult but found: " + table.get("k2-"
            + id);
            assert table.get("k3-" + id).equals("t-1-mult") : "expected to find: t-
1-mult but found: " + table.get("k3-"
            + id);

            userTraceLogger.debug("removeNow");
            table.removeNow("test1-a-" + id);
            table.removeNow("test1-b-" + id);
            table.removeNow("k1-" + id);
            assert table.get("test1-a-" + id) == null : "expected to find: null but 
found: " + table.get("test1-a-" +
            id);
            assert table.get("test1-b-" + id) == null : "expected to find: null but 
found: " + table.get("test1-a-" +
            id);
            assert table.get("k1-" + id) == null : "expected to find: null but 
found: " + table.get("k1-" + id);
            assert table.get("k2-" + id) == null : "expected to find: null but 
found: " + table.get("k2-" + id);
            assert table.get("k3-" + id) == null : "expected to find: null but 
found: " + table.get("k3-" + id);

            userTraceLogger.info("test external storage ended");
```

A.2 API Data Storage

The following table describes the data storage API parameters.

Table 88: API Data Storage Parameters

Parameter Name	Definition	Param Key	Param Value	Param Timeout
public interface StorageProvider<K, V>	Associates the specified value with the specified key in this storage. If the storage previously contained a mapping for the key, the old value is replaced by the specified value. This operation is non-blocking and is performed asynchronously. If it fails, the system logs a warning. This operation uses a default timeout.	The key with which the specified value is associated.	The value to be associated with the specified key.	
boolean put(K key, V value)	Associates the specified value with the specified key in this storage. If the storage previously contained a mapping for the key,	The key with which the specified value is associated.	The value to be associated with the	

Parameter Name	Definition	Param Key	Param Value	Param Timeout
	the old value is replaced by the specified value. This operation is non-blocking and is performed asynchronously. If it fails, the system logs a warning.		specified key.	
boolean put(K key, V value, long timeout, java.util.concurrent.TimeUnit timeUnit)	Associates the specified value with the specified key in this storage. If the storage previously contained a mapping for the key, the old value is replaced by the specified value. This operation is blocking and is performed synchronously. This operation uses a specified timeout.	The key with which the specified value is associated.	The value to be associated with the specified key.	Specifie d timeout
boolean putNow(K key, V value)	Associates the specified value with the specified key in this storage. If the storage previously contained a mapping for the key, the old value is replaced by the specified value. This operation is blocking.	The key with which the specified value is associated.	The value to be associated with the specified key.	
boolean putNow(K key, V value, long timeout, java.util.concurrent.TimeUnit timeUnit))	Associates the specified value with the specified list of keys in this storage. If the storage previously contained a mapping for the key, the old value is replaced by the specified value. This operation is blocking and is performed asynchronously. If it fails, the system logs a warning.	The key with which the specified value is associated.	The value to be associated with the specified key.	The time (in seconds) to keep this element in the storage.

Parameter Name	Definition	Param Key	Param Value	Param Timeout
boolean putAll(List<K > keys, V value)	Associates the specified value with the specified list of keys in this storage. If the storage previously contained a mapping for the key, the old value is replaced by the specified value. This operation is blocking and is performed asynchronously. If it fails, the system logs a warning.	The key with which the specified value is associated.	The value to be associated with the specified key.	.
boolean putAll(List<K > keys, V value, long timeout, java.util.concurrent.TimeUnit timeUnit)	Associates the specified value with the specified list of keys in this storage. If the storage previously contained a mapping for the key, the old value is replaced by the specified value. This operation is blocking until all the values are located inside the external storage. It is performed synchronously.	The key with which the specified value is associated.	The value to be associated with the specified key.	The time (in seconds) to keep this element in the storage.
boolean putAllNow(List<K> keys, V value)	Associates the specified value with the specified list of keys in this storage. If the storage previously contained a mapping for the key, the old value is replaced by the specified value. This operation is blocking until the all the values are located inside the external storage. It is performed synchronously.	The key with which the specified value is associated.	The value to be associated with the specified key.	The time (in seconds) to keep this element in the storage.
boolean putAllNow(Li				The time (in

Parameter Name	Definition	Param Key	Param Value	Param Timeout
st<K> keys, V value, long timeout, java.util.concurrent.TimeUnit timeUnit)				seconds) to keep this element in the storage
V get(K key)	Retrieves an entry in the same way as get, except it does not update or reorder any of the internal constructs. i.e., expiration does not happen, and the entry is not considered as "touched".	The key under which the entry is stored. Return the entry, if it exists, or null if it does not exist.		
V peek(K key)	Removes the mapping of a key from this map, if present. This operation is non-blocking and is performed asynchronously. If it fails, the system logs a warning.	The key that will be removed. Return the removed object in case of success, otherwise returns null.		
void remove(K key)	Removes the mapping of a key from this map, if present. This operation is blocking until the item is removed from the external storage. It is performed synchronously.	The key that will be removed. Return the removed object in case of success, otherwise returns null.		
void removeAll(List<K> keys)		The list of keys to be removed.		

Parameter Name	Definition	Param Key	Param Value	Param Timeout
		Returns true in case of success.		

Appendix B: Supported Application Identifiers

Table 90 describes the Supported Application Identifiers.

Table 89: Supported Application Identifiers

Application Name	Application ID	Vendor ID	Application Type
Base	0	IETF	Authentication and Accounting
NASREQ	1	IETF	Authentication
MobileIPV4	2	IETF	Authentication
BaseAccounting	3	IETF	Accounting
CC	4	IETF	Authentication
EAP	5	IETF	Authentication
SIP	6	IETF	Authentication
Relay	0xFFFFFFFFL	IETF	Authentication and Accounting
Cx	16777216	3GPP	Authentication
Sh	16777217	3GPP	Authentication
Re	16777218	3GPP	Rating
Wx	16777219	3GPP	Authentication
Zn	16777220	3GPP	Authentication
Zh	16777221	3GPP	Authentication
Gmb	16777223	3GPP	Authentication
MM10	16777226	3GPP	Authentication
Pr	16777230	3GPP	Authentication
E4	16777231	ETSI	Authentication
Wa	-1	3GPP	Authentication
Wd	-1	3GPP	Authentication

Application Name	Application ID	Vendor ID	Application Type
Wg	-1	3GPP	Authentication
Wm	-1	3GPP	Authentication
Gi	-1	3GPP	Authentication and Accounting
Rx	16777236	3GPP	Authentication
Gq	16777222	3GPP	Authentication
Rq	16777222	ETSI	Authentication
Gx	16777238	3GPP	Authentication
Tx	16777236	3GPP2	Authentication
Ty	16777237	3GPP2	Authentication
Gxc	16777266	3GPP	Authentication
S9	16777267	3GPP	Authentication
Gxp	16777238	9	Authentication
Gy	4	3GPP	Authentication
Gz	-1	3GPP	Accounting
Rf	3	3GPP	Accounting
Ro	4	3GPP	Authentication
CMS	2	IETF	Authentication
S6b	99999	3GPP	Authentication
SCAP1	19302	193	Accounting
VFDCCA	4	NoVendor	Authentication
TSL	4	NoVendor	Authentication
PS	4	NoVendor	Authentication
S6a	16777251	3GPP	Authentication
S6d	16777251	3GPP	Authentication

Application Name	Application ID	Vendor ID	Application Type
Hd	16777317	3GPP	Authentication
E2	16777231	3GPP	Authentication

Appendix C: Offline Processing Mode

The SDC includes the functionality to write messages offline to .dat files for future use. This message mode – the “degraded” mode – is implemented by configuring a file server to store the messages.

The file server acts as a Diameter peer, where each message received by the file server is parsed. The first AVP defines the path of the degraded file. If the file exists, the message that is contained in the second AVP is saved to this file. If the file doesn’t exist, the file server will create it.

The path name consists of the server peer name and group-id. Each file server can have up to 12 different links with the SDC – one link per peer server.

Each folder can have multiple files with .dat extensions and files with .tmp extensions.

If the file server crashes, when it starts up it looks for all .tmp files and renames them to .crash.

The files are rotated in two cases – when they reach the max number of messages per file or the file was open more the specified timeout. Both of these values are configurable.

The files in the File Server will be located by default in the */home/traffix/FileServer/root/FS1/* folder. When the CPF starts to send requests to the File Server, a new folder with the name of the degraded peer will be created and all requests that are sent to this peer will be located in the */home/traffix/FileServer/root/FS1/PEER-NAME/* folder. It will also create folders with the group-number for each group */home/traffix/FileServer/root/FS1/PEER-NAME/Group-Num/*, and all files will be created based on the peer name and group.

```
The file name format can be configured. By default, it will be
STRFX_FDGPRS_ID0_T(time-stamp)_(host-name-of-the file-server)_GRP(group-
num)_NUM(num-of-messages).dat
```

To configure offline processing:

Configure a file server by performing the following steps:

Go to **Topology** > Remote Peers.

1. Click **Add**. The Add Peer wizard appears.

2. In the **Name** field, set the name for this peer.

3. In the **Protoco**l field, select File.

4.

5. Click **Next**.

6. Set **Primary IP** of the File Server.

7. Set **Primary Port** of the File Server.

8.

9. In the **Split By** field, set the value on which the messages will be divided into groups.

10. In the **Number of Groups** field, set how many group will be needed.

11. In the **FTP Server Name**, select the FTP server for uploading the files from this peer.

12.

13. Click **Finish**.

Go to **Routing** > **Routing**, and configure the file server as either a backup server in case the primary Diameter servers are not available, or as the primary server.

Appendix D: Decision Table Attributes

The following table describes all SDC predefined attributes for various SDC entities which can be used in any of the decision tables, both in the condition fields and the selection configuration. Using the attributes in a decision table is the equivalent of calling the groovy methods getProperty(name) and setProperty(name, value). For example, using session.IS_TRACEBLE in a routing table condition is the equivalent of the groovy method session.isTraceble() from groovy.

The Session entity also supports arbitrary user-defined attributes. You may, for example, configure (=set value) session.IMSI attribute in one of the decision tables, and use the attribute in any of the other decision table's conditions. You may also create and access dynamic properties of the Envelope entity. This entity has no predefined properties. The attributes can be chained. For example: request.SESSION.POOL.NAME.

Table 90: Decision Table Rule Attributes

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
Session	SESSION_ID	String	The session ID	session.getSessionId()	Cannot set	Cannot set
Session	MASTER_SESSION_ID	String	The master session ID if session should be resolved, null otherwise	session.getMasterSession().getSessionId()	Cannot set	Cannot set
Session	CONTEXT_ID	String	The session ID of the master session if exists, otherwise returns the session ID	session.getContextId()	indicates the context to be used in a contextual load	session.setContextId()

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
					balancing policy	
Session	IS_PERSISTENT	Boolean	is session persisted in storage		indicates session persistence in storage	
Session	RELEASE_POLICY	Boolean	deprecated			
Session	IS_TRACEABLE	Boolean	Is session traceable	session.isTraceable()	Marks/unmarks session for tracing	session.setTraceable()
Session	SHOULD_DUMP	Boolean	Should/ should not be dumped to file?	session.shouldDumpMessage()	Indicates writing to a file	session.setShouldDumpMessage()
Session	SHOULD_REPLICATE	Boolean	should be replicated to another site (if SDC site is supported)		indicates session persistence and replication	
Session	DESTINATION_PEER	Peer	Destination peer	session.getDestinationPeer()	Sets the destination peer	session.setDestinationPeer(peer)
Session	DESTINATION_PEER_NAME	Name of destination peer	session.getDestinationPeerName()	Sets the destination peer	session.setDestinationPeerName()	

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
Session	POOL	Pool	The selected pool	session.getPool()	Cannot set	
Session	POOL_NAME	String	Can also use POOL.NAME	session.getPoolName()	Cannot set	
Session	ROUTING_ROW_ID	String	ID of the selected routing row	session.getRoutingRowId()	Cannot set	
Session	SESSION_BINDING_ROW_ID	String	ID of the selected session binding row	session.getSessionBindingRowId()	Cannot set	
Session	IS_STICKY	Boolean	Is routing 'sticks' on session? The default value is True. if False, the session’s routing should be calculated per message	session.isSticky()	Set stickiness mode on session	session.setIsSticky()
Session	AUTOMATIC_RELEASE	Boolean	Should release session automatically in outgoing transformation (In Diameter: 1. STA, 2. CCA with CC-Request-Type TERMINATION/EVENT, 3. ACA	session.shouldAutomaticallyRelease()	Sets automatic release of the session	session.setShouldAutomaticallyRelease()

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
			with Accounting-Record-Type STOP/EVENT)			
Session	IDLE_SESSION_TIMEOUT	Boolean	Should update session timeout upon request arrival	session.shouldRefreshTimeoutOnGet()	Sets refreshing policy	session.setShouldRefreshTimeoutOnGet()
Peer	NAME	String	The peer name	peer.getName()	Cannot set	
Peer	STATE	State.OPEN, State.BUSY, State.OUT_OF_SERVICE, State.CONNECTING, State.BINDING, State.CLOSING, State.CLOSE	The peer state	peer.getState()	Cannot set explicitly	
Peer	PROFILE_NAME	String	The peer profile name	peer.getPeerProfileName()	Cannot set	
Peer	IS_DYNAMIC	Boolean	Is dynamically discovered	peer.isDynamic()	Cannot set	

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
Peer	IS_SERVER	Boolean	Remote server or client	peer.isServer()	Cannot set	
Peer	BINDING_NAME	String	Key for peer binding (inter-protocol session binding)	peer.getBindingName()	Defines peer binding	peer.setBindingName()
Peer	PROTOCOL	Protocol	Remote node protocol (e.g: Protocol.Diameter)	peer.getProtocol()	Cannot set	
Peer	IS_SECURE	Boolean	Is peer secured	peer.isSecure()	Cannot set	
Peer	PENDING_REQUESTS	Integer	The number of pending requests	peer.getPendingRequestsCount()	Cannot set	
Peer	ROUNDTRIP_TIME	Long	Roundtrip time (in millis)	peer.getRoundTripTimeMillis()	Cannot set	
Diameter Peer	REMOTE_REALM	String	The peer’s realm as published by the other party	peer.getMetaData().getRealmFromCapabilities()	Cannot set	
Diameter Peer	REMOTE_HOST	String	The peer’s host as published by the other party	peer.getMetaData().getHostFromCapabilities()	Cannot set	
Diameter Peer	LOCAL_REALM		The peer's realm as configured by its	peer.getMetaData().getLo	Cannot set	

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
			domain or its profile	calConfiguredRealm()		
Diameter Peer	LOCAL_HOST		The peer's host as configured on its domain or in its profile	peer.getMetaData().getLocalConfiguredHost()	Cannot set	
Diameter Peer	SRR_VERSION	String	The peer’s SRR version	peer.getProperty("SRR_VERSION")	Cannot set	
Pool	NAME	String	The pool’s name	pool.getName()	Cannot set	
Pool	STATE	State.OPEN, State.CLOSE, State.OUT_OF_SERVICE	The pool’s state	pool.getState()	Cannot set	
Pool	SIZE	Integer	The number of active servers	pool.size()	Cannot set	
Message	NAME	String	The massage’s name	message.getName()	Cannot set	
Message	LENGTH	Integer	The message’s length	message.getMessageLength()	Cannot set	
Message	IS_REQUEST	Boolean	Is a request	message.isRequest()	Cannot set	

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
Diameter message	VERSION	Byte	The Diameter version	message.getVersion()	Cannot set	
Diameter message	IS_ERROR	Boolean	Is a Diameter protocol error notification	message.isError()	Cannot set	
Diameter message	IS_PROXIABLE	Boolean	Is the request proxiable	message.isProxiable()	Cannot set	
Diameter message	IS_RETRANSMITTED	Boolean	Is the message potentially retransmitted	message.isReTransmitted()	Cannot set	
Diameter message	COMMAND_CODE	Integer	The message’s command code	message.getCommandCode()	Cannot set	
Diameter message	APPLICATION_ID	Long	The application’s ID	message.getApplicationId()	Cannot set	
Diameter message	HOP_BY_HOP_ID	Long	The hop-by-hop ID	message.getHopIdentifier()	Cannot set	
Diameter message	END_TO_END_ID	Long	The end-to-end ID	message.getEndToEndIdentifier()	Cannot set	
Diameter message	IMSI	String	The Subscription-Id-Data when type is END_USER_IMSI	message.getImsi()	Cannot set	

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
Diameter message	MSISDN	String	The Subscription-Id-Data when type is END_USER_E164	message.getMsisdn()	Cannot set	
Content	NAME	String	The content unit's name	content.getName()	Cannot set	
Diameter AVP	CODE	Integer	The AVP's code	avp.getCode()	Cannot set	
Diameter AVP	V_FLAG	Boolean	The vendor flag	avp.isVendorId()	Cannot set	
Diameter AVP	M_FLAG	Boolean	Is the flag mandatory?	avp.isMandatory()	Cannot set	
Diameter AVP	P_FLAG	Boolean	Is the flag protected?	avp.isEncrypted()	Cannot set	
Diameter AVP	LENGTH	Integer	The AVP's length	avp.getLength()	Cannot set	
Diameter AVP	VENDOR_ID	Long	The vendor ID	avp.getVendorId()	Cannot set	
RADIUS Message	CODE	Integer	The message's code	message.getCommandCode()	Cannot set	
RADIUS Message	IDENTIFIER	Integer	The message's identifier	message.getHopIdentifier()	Cannot set	
RADIUS Message	LENGTH	Integer	The message's length	message.getLength()	Cannot set	

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
RADIUS Message	AUTHENTICATOR	Byte Array	The message's authenticator	message.getAuthenticator()	Cannot set	
RADIUS Attribute	TYPE	Integer	The attribute's type	attribute.getAttributeType()	Cannot set	
RADIUS Attribute	LENGTH	Integer	The attribute's length	attribute.getAttributeLength()	Cannot set	
RADIUS Attribute	VENDOR_ID	Integer	Vendor ID of the attribute	attribute.getVendorId()	Cannot set	
RADIUS Attribute	TAG	Byte	Tag attribute	attribute.getTag()	set tag attribute	attribute.setTag()
Stack	NAME	Name of node	stack.getName()	cannot set	Cannot set	
Stack	STATE	state of stack: State.OPEN, State.CLOSE	stack.getState()	cannot set	cannot set	
Stack	UID	Instance ID of node		cannot set	Cannot set	
Stack	CPF_GROUP_NAME	Group name of node		cannot set	Cannot set	

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
HTTP Message	VERSION	String	The HTTP Version identifier	message.getProperty("VERSION")	Cannot set	Cannot set
HTTP Message	<Header Name>	String	Gets any header content from an HTTP message	message.get(<Header Name>)	Cannot set	Cannot set
HTTP Request	METHOD	String	The HTTP Method’s name (Get, Post etc)	message.getProperty("METHOD")	Cannot set	Cannot set
HTTP Request	URI	String	The HTTP URI Field	message.getProperty("URI")	Cannot set	Cannot set
HTTP Answer	STATUS_CODE	Integer	The HTTP Answer’s response code	message.getProperty("STATUS_CODE")	Cannot set	Cannot set
HTTP Answer	REASON_PHRASE	String	The HTTP Answer’s reason description	message.getProperty("REASON_PHRASE")	Cannot set	Cannot set

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
SS7 Message	OPERATION_CODE	Integer	TCAP Component (usually GSM-MAP) command code	message.getProperty("OPERATION_CODE")	N/A	Cannot set
SS7 Message	ERROR_CODE	Integer	TCAP Component (usually GSM-MAP) error code	message.getProperty("ERROR_CODE")	N/A	Cannot set
SS7 Message	DESTINATION_	Boolean	Shall message be routed by SCCP	message.getProperty("DE	N/A	message.setProperty("D

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
	ROUTE_ON_GT		layer according to the Global Title	STINATION_ROUTE_ON_GT")		ESTINATION_ROUTE_ON_GT")
SS7 Message	DESTINATION_GT_ADDRESS	String	The called global title number	message.getProperty("DESTINATION_GT_ADDRESS")	N/A	message.setProperty("DESTINATION_GT_ADDRESS")
SS7 Message	DESTINATION_GT_TRANSLATION_TYPE	Integer	The translation type attribute of the destination global title	message.getProperty("DESTINATION_GT_TRANSLATION_TYPE")	N/A	message.setProperty("DESTINATION_GT_TRANSLATION_TYPE")
SS7 Message	DESTINATION_GT_NUMBERING_PLAN	Integer	The numbering plan attribute of the destination global title	message.getProperty("DESTINATION_GT_NUMBERING_PLAN")	N/A	message.setProperty("DESTINATION_GT_NUMBERING_PLAN")
SS7 Message	DESTINATION_GT_ENCODING_SCHEME	Integer	The encoding scheme attribute of the destination global title	message.getProperty("DESTINATION_GT_ENCODING_SCHEME")	N/A	message.setProperty("DESTINATION_GT_ENCODING_SCHEME")

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
SS7 Message	DESTINATION_GT_NATURE_OF_ADDRESS_IND	Integer	The NOA (nature of address) attribute of the destination global title	message.getProperty("DESTINATION_GT_NATURE_OF_ADDRESS_IND")	N/A	message.setProperty("DESTINATION_GT_NATURE_OF_ADDRESS_IND")
SS7 Message	DESTINATION_GT_INDICATOR	Integer	The GT Indicator attribute of the destination global title	message.getProperty("DESTINATION_GT_INDICATOR")	N/A	message.setProperty("DESTINATION_GT_INDICATOR")
SS7 Message	ORIGINATION_ROUTE_ON_GT	Integer	The GT Indicator attribute of the destination global title	message.getProperty("ORIGINATION_ROUTE_ON_GT")	N/A	message.setProperty("ORIGINATION_ROUTE_ON_GT")
SS7 Message	ORIGINATION_GT_ADDESS	String	The calling global title number	message.getProperty("ORIGINATION_GT_ADDESS")	N/A	message.setProperty("ORIGINATION_GT_ADDESS")
SS7 Message	ORIGINATION_GT_TRANSLATION_TYPE	Integer	The translation type attribute of the origin global title	message.getProperty("ORIGINATION_GT_TRANSLATION_TYPE")	N/A	message.setProperty("ORIGINATION_GT_TRANSLATION_TYPE")

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
SS7 Message	ORIGINATION_GT_NUMBERING_PLAN	Integer	The numbering plan attribute of the origin global title	message.getProperty("ORIGINATION_GT_NUMBERING_PLAN")	N/A	message.setProperty("ORIGINATION_GT_NUMBERING_PLAN")
SS7 Message	ORIGINATION_GT_ENCODING_SCHEME	Integer	The encoding scheme attribute of the origin global title	message.getProperty("ORIGINATION_GT_ENCODING_SCHEME")	N/A	message.setProperty("ORIGINATION_GT_ENCODING_SCHEME")
SS7 Message	ORIGINATION_GT_NATURE_OF_ADDRESS_IND	Integer	The nature of the address (NOA) attribute of the origin global title	message.getProperty("ORIGINATION_GT_NATURE_OF_ADDRESS_IND")	N/A	message.setProperty("ORIGINATION_GT_NATURE_OF_ADDRESS_IND")
SS7 Message	ORIGINATION_GT_INDICATOR	Integer	The GT Indicator attribute of the origin global title	message.getProperty("ORIGINATION_GT_INDICATOR")	N/A	message.setProperty("ORIGINATION_GT_INDICATOR")
SS7 Message	ORIGINATION_SSN	Integer	The Origin Subsystem Number	message.getProperty("ORIGINATION_SSN")	N/A	message.setProperty("ORIGINATION_SSN")

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
SS7 Message	DESTINATION_SSN	Integer	The Destination Subsystem Number	message.getProperty("DESTINATION_SSN")	N/A	message.setProperty("DESTINATION_SSN")
GTP' Message	CODE	Integer	The GTP' message command code	message.getProperty("CODE")	N/A	message.setProperty("CODE")
GTP' Message	IDENTIFIER	Object	The GTP' message sequence ID	message.getProperty("IDENTIFIER")	N/A	message.setProperty("IDENTIFIER")
GTP' Message	LENGTH	Integer	The GTP' message length	message.getProperty("LENGTH")	N/A	N/A
GTP' Message	VERSION	Integer	The GTP' message version ID	message.getProperty("VERSION")	N/A	N/A
GTP' Message	ORIGIN_PEER	String	The GTP' message's origin peer address	message.getProperty("ORIGIN_PEER")	N/A	message.setProperty("ORIGIN_PEER")
LDAP Message	OPERATION	Integer	The LDAP operation code	message.getProperty("OPERATION")	N/A	N/A
LDAP Message	COMMAND_CODE	Integer	The LDAP operation code	message.getProperty("COMMAND_CODE")	N/A	N/A

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
LDAP Request	DN	Integer	The DN attribute of LDAP request	message.getProperty("DN")	N/A	N/A

Appendix E: Configuring LDAP Authentication

You can configure the SDC to recognize external LDAP users for SDC user login. This is done by accessing an external LDAP server. This access is authenticated by updating the security file (*ldap/applicationContext-security.xml*) and by configuring the LDAP parameters in the *ldap-config.properties* file.

To enable LDAP user login:

Note: Steps 1-3 must be done on the two master Installers.

1. Go to the */srv/salt/5.2 <version number>/webui* folder.
2. Copy the *ldap/applicationContext-security.xml* to *applicationContext-security.xml* under the current *webui* folder.
3. Open the *ldap/ldap-config.properties* file to change the default parameters.

To enable user login using an external LDAP server:

4. Edit the following attributes (in the table below) in the *ldap-config.properties* file (as applicable):

Note: You must change the default parameters, unless noted for a specific parameter. This file has the following limitations:

Spaces are not allowed at the end of a row.

Certain characters are defined in the file format with specific attributes. Therefore, when the value contains one or more of the following characters, preface it with the ‘\’ symbol, as follows:

Instead of ‘=’, use ‘\=’.

Instead of ‘:’, use ‘\:’.

Instead of ‘\’, use ‘\\’.

Table 91: LDAP Attributes

Attribute	Description	Mandatory	Example
url	The address, port, and root directory of the LDAP server against which the authentication will be performed.	Yes	ldap\://ldap-ca.lab.traffixsystems.com\:389 Note: when the SSL encryption method is used, the value will be: ldaps\://ldap-ca.lab.traffixsystems.com\:636
second.url	A second for the LDAP server, for fail-over scenarios.	No	ldap\://ldap-ca.lab.traffixsystems.com\:636
ldap.base	The LDAP base directory on the LDAP server	Yes	dc\=lab,dc\=traffixsystems,dc\=com
manager.dn	The LDAP server username.	Yes	cn\=Manager,dc\=lab,dc\=traffixsystems,dc\=com
password	The LDAP server password.	Yes	ENC(wTkETma1KbgAFlJb9RmY8ek34bX4WT4m)
def.group.search.base	The base DN under which the LDAP integration should look for matches for the user DN.	No	ou\=groups Note: When empty, the search is performed from the LDAP root

group.search.filter	The attribute type and value used by the search filter in the group.search.base. The filter is either by the user DN (0) or by the username (1).	Yes	memberUid\={1}
			Default: uniqueMember={0}

group.role.attribute	The attribute to check for matching entries	Yes	cn

Attribute	Description	Mandatory	Example
user.search.base	The base directory under which the LDAP integration should look for matches for the user’s id.	No	ou\=users Note: When empty, the search is performed from the LDAP root
user.search.filter	The LDAP search filter used to match the user’s id to an attribute of an entry located under defined base directory.	Yes	(uid\={0})
search.subtree	Defines if searches can also performed on sub-trees in the LDAP directory	Yes	true
role.prefix	The prefix that will be added to the value found in group-role-attribute. This is needed to create a Spring Security authority object.	Yes	ROLE_ Note: There is no need to change this default value.
password.encoder	The password encryption	No	shaPasswordEncoder
role.user.read	Groups of users with read only permissions.	Yes	users
role.expert.execute	Groups of users with execute permissions	Yes	admin, expert
role.rnd.manage	Groups of users with permissions to manage engineering scripts.	Yes	admin
authenticationStrateg y	The authentication processing behavior.	Yes	default – defines clear text and SSL startTLS – defines the start TLS behavior SSL – defines SSL with a certificate

Attribute	Description	Mandatory	Example
trust.store	The location of the security certificate.	Yes Note: Only for the startTLS and SSL authentication strategies	C\:\\Temp\\sslkey.jks
trust.store.password	The password of the security certificate.	Yes Note: Only for the startTLS and SSL authentication strategies	ENC(W4WStHUig4GJkm5QR2PNacoFQb8Fcbu1)

4. Run the following command on one master Installer:

```
salt "*" state.highstate
```

Removing LDAP Authentication

After allowing LDAP user login, you can revert the Web UI authentication to the original SDC login configuration.

To disable LDAP user login:

5.

Note: Steps 1-2 must be done on the two master Installers.

Go to the */srv/salt/5.2 <version number>/webui* folder.

Copy the *default/applicationContext-security.xml* to *applicationContext-security.xml* under the current *webui* folder.

6. Run the following command on one master Installer:

```
salt "*" state.highstate
```

7.

Glossary

The following tables list the common terms and abbreviations used in this document.

Table 92: Common Terms

Term	Definition
Answer	A message sent from one Client/Server Peer to the other following a request message
Client Peer	A physical or virtual addressable entity which consumes AAA services
Data Dictionary	Defines the format of a protocol’s message and its validation parameters: structure, number of fields, data format, etc.
Destination Peer	The Client/Server peer to which the message is sent
Geo Redundancy	A mode of operation in which more than one geographical location is used in case one site fails
Master Session	The session for which the routing selection is performed based on the routing rules (Slave Sessions are applied with routing rules inherited from the Master Session)
Orchestrator	A workflow management solution to automate the creation, monitoring, and deployment of resources in your environment
Origin Peer	The peer from which the message is received
Pool	A group of Server Peers
QCOW2	A file format for disk image files
RADIUS	Remote Authentication Dial In User Service
REST	Representation of a resource between a client and server **(**Representational **S**tate Transfer)
Request	A message sent from one Client/Server peer to the other, followed by an answer message

Term	Definition
RPM	RPM Package Manager
Salt-API	Manages and communicates between an Orchestrator and network master and minion servers
SDC Site	The entire list of entities working in a single site
Server Peer	A physical or virtual addressable entity which provides AAA services
Session	An interactive information interchange between entities
Slave (Bound) Session	A session which inherits properties from a master session
Transaction	A request message followed by an answer message
Tripo	Session data repository
vCenter	Vmware Virtual Infrastructure tool for centralized management of multiple hypervisors and enabling functionalities
Virtual Server	A binding point used by SDC to communicate with the Remote Peers (Clients and Servers)

Table 93: Abbreviations

Term	Definition
AAA	Authentication, Authorization and Accounting
ACL	Access Control List
AF	Application Function
API	Application Programming Interface
AVP	Attribute Value Pair
CLI	Command Line Interface
CPF	Control Plane Function
DEA	Diameter Edge Agent

Term	Definition
DRA	Diameter Routing Agent
EMS Site	Element Management System Site
FEP-In	In-Front End Proxy
FEP-Out	Out-Front End Proxy
HA	High Availability
HSS	Home Subscriber Server
HTTP	Hypertext Transfer Protocol
IaaS	Infrastructure as a Service
IMS	IP Multimedia Subsystem
JMS	Java Message Service
KPI	Key Performance Indicator
LDAP	Lightweight Directory Access Protocol
LTE	Long Term Evolution
MME	Mobility Management Entity
NGN	Next Generation Networking
Node	Physical or virtual addressable entity
OAM	Operation, Administration and Maintenance
OCS	Online Charging System
OVF	Open Virtualization Format
PCEF	Policy and Charging Enforcement Function
PCRF	Policy and Charging Rules Function
PLMN	Public Land Mobile Network
SCCP	Signaling Connection Control Part

Term	Definition
SCTP	Stream Control Transmission Protocol
SDC	Signaling Delivery Controller
SNMP	Simple Network Management Protocol
SS7	Signaling System No. 7
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
UE	User Equipment
URI	Universal Resource Identification.
VIP	Virtual IP
VM	Virtual Machine
VNFC	Virtualized Network Function Component
VPLMN	Visited Public Land Mobile Network
Web UI	Web User Interface
WS	Web Service