Signaling Delivery Controller

User Guide

4.4

Catalog Number: RG-015-44-22 Ver.2

Publication Date: June 2015

Proprietary and Confidential Information of F5 Networks

Legal Information

Copyright

© 2005-2015 F5 Networks, Inc. All rights reserved.

F5 Networks, 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 Networks, OpenBloX, OpenBloX (design), Rosetta Diameter Gateway, Signaling Delivery Controller, SDC, Traffix, and Traffix [DESIGN] are trademarks or service marks of F5 Networks, 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 Networks. The information in this document may be changed at any time without notice.

About F5 Networks

F5 Networks (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-015-44-22 Ver.2

Publication Date: June 2015

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.

Document History

Revision Number	Change Description	Change Location
June 2015 – Ver. 2	Description and names of tabs in Enabling EU Local Breakout for Diameter and SS7 peer profiles. Removed note about enabling Threshold Management alarms for rate limits. Updated Syslog Daemon configuration section. Removed Syslog display filter. Removed SIP protocol configurations. Update types of possible result codes when selecting a Discard with Answer Read Limit Policy, added note for Peer Rate Limits. Added note about enabling session logs and updated Session Output Log File table. Added section about configuring user properties on a site level. New Add Pool screenshot with Alarms tab and Alarms tab step.	See *Diameter Peer Profile* and *SS7 Peer Profile;* 🌐 *Threshold Management; Defining Syslog Daemon Addresses; System History Status, Configuring the Incoming Traffic Rate Limits, Enabling the Session Life Cycle and Session Error Logs, Configuring a Site's User Properties. Adding a New Pool,* 🌐 *System View,* Accessing *the SDC/EMS Web UI, Trap Descriptions; SDC Node KPIs*

Revision Number	Change Description	Change Location
	Auto Refresh button in Monitoring, System View screen. Trademark text updated. Updated recommended Mozilla Firefox version Added CpfNmsCollectingStatisticsFailureClear and updated other SNMP traps. Updated SDC Node KPIs.	

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

1. Working with the SDC 1
1.1 Working with the Web UI 1
.2 Getting to Know the SDC/EMS Web UI 3
2.1 Accessing the SDC/EMS Web UI 3
2.2 Logging in to the SDC/EMS Web UI 4
2.3 Using the SDC/EMS Web UI 4
2.3.1 The Menu Bar 5
2.3.2 The Tab Bar 6
2.3.3 The Navigation Pane 7
2.3.3.1 The Navigation Pane Filter 7
2.3.4 Common Actions 7
2.3.5 Keyboard Navigation 9
2.3.6 SDC Decision Tables 10
2.3.6.1 Adding Rule Attributes 11
2.3.6.2 Defining Rule Attribute Values 12
2.3.6.3 Defining Rule Actions and Configurations 13
3. Configuring the SDC Topology 14
3.1 🌐 Topology View 14
3.1.1 Peer Profiles 14
3.1.2 Global Properties 15
3.1.3 Specific Site Settings 15
3.1.3.1 SDC Components 15
3.1.3.2 Virtual Servers 17
3.1.3.3 Remote Peers 17
3.1.3.4 Pools 18
3.1.3.5 Access Control List 18
3.1.4 The Control Plane Traffic Flow – SDC’s Services 19
3.1.5 Topology Architecture 19
3.2 Configuring the Topology 20
3.2.1 Configuring Peer Profiles 21
3.2.1.1 Viewing the List of Peer Profiles 21
3.2.1.2 Adding a Peer Profile 22
3.2.2 Association Rules 38
3.2.2.1 Adding a New Association Rule Attribute 38
3.2.2.2 The Association Rules 40
3.2.3 Configuring a Site's User Properties 41
3.2.4 Configuring the SDC Components 42
3.2.4.1 Viewing the Internal Connections 46
3.2.5 Configuring Virtual Servers 47
3.2.5.1 Viewing the Virtual Servers 47
3.2.5.2 Adding a New Virtual Server 48
3.2.6 Configuring Remote Peers 51
3.2.6.1 Viewing the List of Remote Peers 52
3.2.6.2 Adding a New Remote Peer 54

3.2.6.3 Removing a Remote Peer 63
3.2.7 Configuring Pools 63
3.2.7.1 Viewing a List of Pools 64
3.2.7.2 Adding a New Pool 65
3.2.7.3 Assigning a Load Balancing Policy 69
3.2.7.4 Editing a Pool 80
3.2.7.5 Removing a Pool 80
3.2.8 Configuring the Access Control List 81
3.2.9 Health Monitoring 83
3.2.10 Error Detection Monitor 84
3.2.10.1 Timeout Monitor 84
3.2.10.2 Response Based Monitor 84
3.2.10.3 Setting an Error Detection Monitor Parameters 85
3.2.10.4 Custom Service Availability Monitor 85
3.2.10.5 Adding a Service Availability Health Monitor 86
3.3 Site Replication 88
3.4 Default Transport Configuration 90
3.4.1 Configuring Rate Limits 94
3.4.1.1 Configuring the Incoming Traffic Rate Limits 95
3.4.1.2 Configuring the Outgoing Traffic Rate Limits 96
3.4.2 Replicating Session Data 97
3.5 Licensing the VIPs 98
3.5.1 Adding a New License Key 98
3.5.1.1 The License Key's Structure 98
3.5.2 Removing a License Key 99
4. Configuring the SDC Flow Management 100
4.1 🌐 Dictionary 101
4.2 🌐 External Lookup Management 102
4.2.1 Disabling External Lookup 105
4.2.2 Removing an External Lookup Data Source 106
4.3 🌐 Session Management 106
4.3.1 Configuring Session Binding Rules 107
4.3.2 Adding a Session Binding Attribute 108
4.3.3 Adding a Session Binding Rule 109
4.3.3.1 Defining Binding Keys 111
4.3.3.2 Selecting a Defined Binding Key 111
4.3.3.3 Defining Session Properties 112
4.3.3.4 Configuring Session Life-Cycle Scripts 114
4.3.3.5 Configuring the External Lookup Script 118
4.4 Routing Mechanism 121
4.4.1 Configuring a Routing Rule 122
4.4.1.1 Defining Routing Rule Attributes 123
4.4.1.2 Defining Routing Rules 124
4.5 Transformation 139
4.5.1.1 Creating a New Transformation 140
4.5.1.2 Adding a Transformation Rule Attribute 140
4.5.1.3 Adding a New Transformation Rule 142
4.5.1.4 Adding a Transformation Script 143

- 4.6 SDC Life Cycle Scripts ... 144

5. Monitoring the SDC ... 146

- 5.1 🌐 Threshold Management ... 146
- 5.2 SDC Dashboard ... 147
  - 5.2.1 Statistics Filtering Settings ... 161
  - 5.2.2 Statistic Data Collection ... 163
  - 5.2.3 Configuring the Data Collection ... 164
- 5.3 EMS Dashboard ... 165
- 5.4 🌐 Reports ... 166
  - 5.4.1 SDC Node KPIs ... 166
  - 5.4.2 Remote Peer KPIs ... 171
  - 5.4.3 Transactions KPIs ... 174
  - 5.4.4 TDR Dashboard ... 176
  - 5.4.5 Transaction Data Records ... 177
  - 5.4.6 Traced Messages ... 180
  - 5.4.7 Session KPIs ... 181
  - 5.4.8 Repository KPIs ... 183
- 5.5 SNMP Traps ... 185
  - 5.5.1 Defining the SNMP Target ... 186
  - 5.5.2 SNMP Dilution Manager ... 187
    - 5.5.2.1 Trap Descriptions ... 187
    - 5.5.2.2 Configuring the Dilution Manager ... 191
  - 5.5.3 SNMP Logs ... 192
  - 5.5.4 Monitoring SNMP Traps ... 192
    - 5.5.4.1 Trap Viewer ... 192
- 5.6 🌐 System View ... 193
- 5.7 🌐 System Performance ... 196
  - 5.7.1 Enabling the Session Life Cycle and Session Error Logs ... 197
- 5.8 🌐 Host Performance ... 200
- 5.9 System History Status ... 202
- 5.10 Logging and Syslog ... 203
  - 5.10.1 Setting the Log Levels ... 204
  - 5.10.2 Defining Syslog Daemon Addresses ... 206
  - 5.10.3 Log File Size Control ... 207
- 5.11 🌐 Tracing ... 207
  - 5.11.1 Configuring a Tracing Rule ... 208
  - 5.11.2 Defining Tracing Rule Attributes ... 208
  - 5.11.3 Adding a Tracing Rule ... 208

6. Managing the SDC ... 210

- 6.1 🌐 Restoring Previous Configurations ... 210
  - 6.1.1 Auditing ... 210
  - 6.1.2 Backup & Restore ... 213
- 6.2 🌐 User Management ... 214
- 6.3 FTP servers ... 217

Appendix A: User Data Storage ... 218

A.1 Implementation Example ... 220

A.2 API Data Storage 221
Appendix B: Supported Application Identifiers 226
Appendix C: Offline Processing Mode 228
Appendix D: Decision Table Attributes 230
Appendix E: Configuring LDAP Authentication 248
Glossary 254

List of Figures

Figure 1: SDC Web UI's Interface 5
Figure 2: The Navigation Pane Filter 7
Figure 3: An Element in Focus 10
Figure 4: Topology View 14
Figure 5: Network Architecture 16
Figure 6: SDC, Client and Server 16
Figure 7: SDC Network Topology 20
Figure 8: Peer Profiles 21
Figure 9: Add Peer Profile Wizard 23
Figure 10: Peer Profile Configuration 24
Figure 11: Diameter Configuration 26
Figure 12: Peer Profile Handshake 30
Figure 13: Association Rule Attributes 39
Figure 14: Attribute Types 40
Figure 15: Association Rules 41
Figure 16: SDC Components 42
Figure 17: Remote Peers 52
Figure 18: Add Server Peer Window 55
Figure 19: Pools 64
Figure 20: Add Pool 66
Figure 21: By Precedence Policy 70
Figure 22: Round Robin Policy 71
Figure 23: Weighted Round Robin Policy 72
Figure 24: Fastest Response Time Policy 73
Figure 25: Queue Size Ratio Policy 74
Figure 26: Load Based Policy 75
Figure 27: Contextual Policy 76
Figure 28: Weighted Contextual Policy 77
Figure 29: External Policy 78
Figure 30: Error Events in a Measuring Interval 85
Figure 31: Site Replication 89
Figure 32: Replication Targets 89

Figure 33: Basic Traffic Flow between the SDC and Networks 94
Figure 34: License Key 99
Figure 35: SDC Internal Flow Logic 100
Figure 36: Data Dictionary 102
Figure 37: Add External Lookup 103
Figure 38: Session Management 108
Figure 39: On Session Create Script 114
Figure 40: On Session Update Script 115
Figure 41: On Session Release Script 117
Figure 42: External Lookup Script 119
Figure 43: Rule Attributes 123
Figure 44: SDC Diameter Identity Persistence 129
Figure 45: Transformation Rule Attributes 141
Figure 46: Post-Routing Transformation Rules 142
Figure 47: SDC Life Cycle Scripts 145
Figure 48: Threshold Management 147
Figure 49: SDC Dashboard Statistics Graphs 148
Figure 50: Typical Line Bar 148
Figure 51: Statistics Filtering Settings 162
Figure 52: Statistics Types 163
Figure 53: EMS Dashboard Display 165
Figure 54: SDC Node KPIs Reports 167
Figure 55: Remote Peer KPI Reports 171
Figure 56: Transaction KPIs 175
Figure 57: Reports>TDR Dashboard – Number of Messages view 177
Figure 58: Reports>Transaction Data Records 178
Figure 59: Traced Messages 180
Figure 60: Traced Messages – 4 Messages 181
Figure 61: Repository KPIs Reports 184
Figure 62: SNMP Targets 186
Figure 63: Alarm Properties 192
Figure 64: Trap Viewer 193
Figure 65: System View 194
Figure 66: System Status Table 195
Figure 67: System Performance 196
Figure 68: Host Performance 201
Figure 69: Host Info 201
Figure 70: System History Status 203
Figure 71: Tracing Rules 209
Figure 72: Audit 212
Figure 73: Backup & Restore 213

Figure 74: Snapshot Description 214
Figure 75: User Management 215
Figure 76: Add User........ 216
Figure 77: User Data Storage 219

List of Tables

Table 1: Conventions III
Table 2: The Menu Bar........ 5
Table 3: The Tab Bar 6
Table 4: Common Actions 8
Table 5: Keyboard Navigation........ 9
Table 6: SDC Network Topology Legend........ 20
Table 7: Peer Profile’s Properties 21
Table 8: Request and/or Answer Scripts Parameters........ 30
Table 9: SDC Components........ 42
Table 10: General SDC Component’s Properties 44
Table 11: Diameter SDC Component’s Properties........ 45
Table 12: SS7 SDC Component’s Properties 45
Table 13: SDC Component’s User Properties 46
Table 14: Internal Connections........ 46
Table 15: Virtual Server Properties........ 47
Table 16: Remote Peer's Properties 52
Table 17: Pool’s Properties 64
Table 18: External Script Parameters........ 79
Table 19: Statistic Data Functions........ 85
Table 20: Health Monitor Condition Script Parameters 87
Table 21: Health Monitor Check Script Parameters 87
Table 22: Socket Defaults........ 91
Table 23: Rate Limit 95
Table 24: License Key Properties 99
Table 25: SDC Flow Logic Legend........ 100
Table 26: Lookup Script Parameters........ 105
Table 27: Session Binding Methods........ 109
Table 28: On Session Create Script Parameters 114
Table 29: On Session Update Script Parameters 116
Table 30: On Session Release Script Parameters........ 118
Table 31: External Lookup Script Parameters........ 119
Table 32: Action Descriptions 124
Table 33: Routing Rule Configuration Parameters 126
Table 34: Check Error in Answer Returned Value........ 131
Table 35: Check Error in Answer Script Parameters........ 132

Table 36: Handle Server Error Script Parameters 133
Table 37: Handle Client Error Script Parameters 134
Table 38: Handle Locally Script Parameters 135
Table 39: Handle Locally Script Parameters 135
Table 40: Redirect Script Parameters 136
Table 41: Reject Script Parameters 137
Table 42: Create Message Locally Script Parameters 138
Table 43: Transformation Script Parameters 143
Table 44: SDC Life Cycle Script Parameters 145
Table 45: Dashboard Graphs 149
Table 46: Transaction’s Roundtrip Time Stamps 164
Table 47: EMS Dashboard Graphs 165
Table 48: SDC Node KPI Report Types 168
Table 49: Remote Peer KPI Report Types 172
Table 50: Remote Node Event Result Codes 175
Table 51: TDR Collected Data 178
Table 52: Session KPI Report Types 182
Table 53: Repository KPI Report Types 184
Table 54: SNMP Targets Table 186
Table 55: SDC Generated Traps 187
Table 56: SNMP Dilution Manager Table 191
Table 57: Trap Viewer Table 193
Table 58: System View 194
Table 59: System Status Table 195
Table 60: System Performance Graphs 197
Table 61: Life Cycle Events Written to the Session Output Log File• 197
Table 62: Host Performance Graphs 202
Table 63: System Status Table 203
Table 64: Log Detail Level 204
Table 65: Customized Log Level Categories 205
Table 66: Syslog Addresses 207
Table 67: Log File Size 207
Table 68: Audit Entries Properties 212
Table 69: Backup Snapshot Properties 213
Table 70: User Type Privileges 215
Table 71: SDC Users 215
Table 72: API Data Storage Parameters 221
Table 73: Supported Application Identifiers 226
Table 74: Decision Table Attributes 230
Table 75: LDAP Attributes 248
Table 76: Terms and Abbreviations 254

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 is accessible through a Web UI. In addition, certain functionalities are also available through 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*.

1.1 Working with the Web UI

There are two available Web UI models: SDC and EMS (Element Management System).

- The SDC Web UI is used for single or multiple SDC site deployments.
- The EMS Web UI is used when you have an EMS deployment. An EMS deployment allows management of multiple SDC sites with a simple global user interface. With EMS, you may perform global configurations and 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: For maximum benefit of the SDC solution, an EMS deployment with an EMS Web UI is recommended.

Throughout the User Guide, certain procedures are noted as EMS only (marked with this icon: 🌐) as the EMS Web UI allows you to perform certain actions, such as global configurations and viewing certain reports, that are not available with the SDC Web UI.

You can do the following actions with the Web UI:

- *Configuring the SDC Topology*
- *Configuring the SDC Flow Management*

- *Monitoring the SDC*
- *Managing the SDC*

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.

Note: The procedures described in this guide follow the installation procedures described in the *F5 SDC Installation Guide*. If you have not yet performed the installation procedures, refer to the *F5 SDC Installation Guide*.

The SDC and EMS Web UIs have a very similar "look and feel." There are some actions, however, that are different in the SDC and EMS Web UI, such as the **Dashboard** tab in the tab bar, and there are some that are only available in the EMS Web UI, such as the **Reports** tab in the tab bar. Those actions that are different or only available for EMS are marked accordingly (🌐).

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:

1. Launch a web browser.

   Note: The SDC/EMS Web UI supports IE9 and Mozilla Firefox 14.0.1. The recommended web browser to view SDC/EMS graphs is Mozilla Firefox 37.

2. Enter the following HTTP path:

   http://<IP address>:8080/MgmtConsole/MgmtConsole.html in the browser's address line (the IP address that is defined for the Web UI resource 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:

To log in to SDC/EMS Web UI:

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

You will be required to enter your username and password again if you take a break longer than 60 minutes from using SDC Web UI.

To access the user interface with a different user name, change your password or log out:

1. From the login toolbar, click **Switch User**/ **Change Password/ Log Out**:

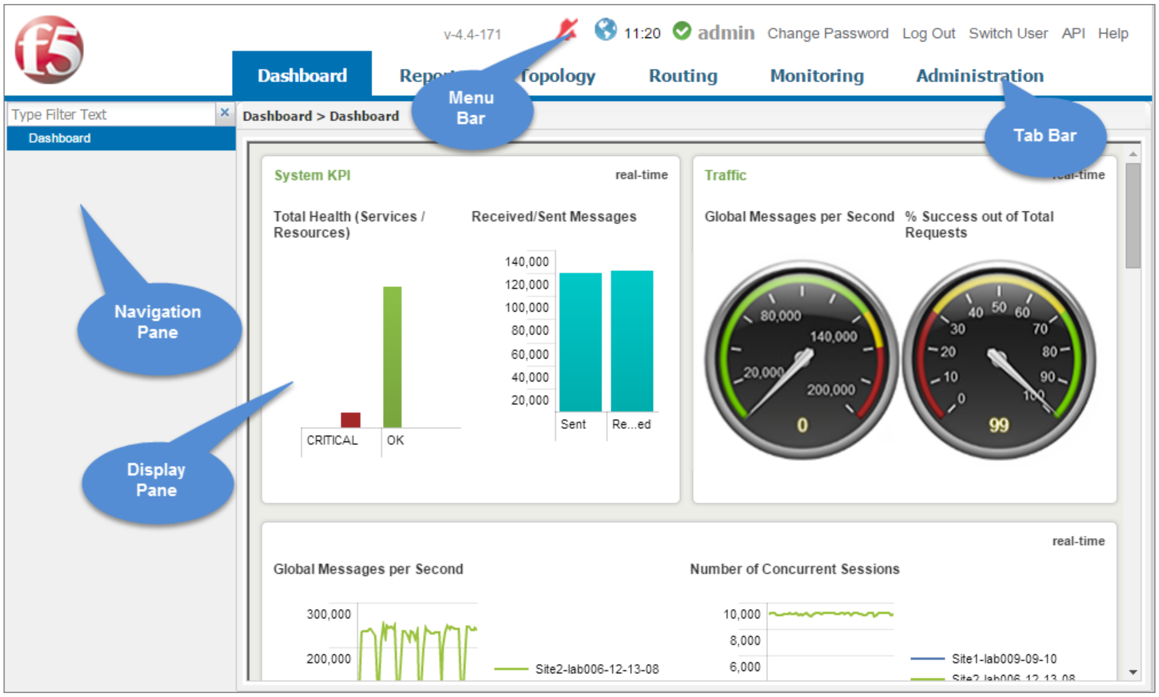
Note: For additional information on users and user roles, see *User Management.*

2.3 Using the SDC/EMS Web UI

The interface is comprised of the following areas:

- *The Menu Bar*
- *The Tab Bar*
- *The Navigation Pane*
- The display pane

Figure 1: SDC Web UI's Interface



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	Enables you to view generated traps in the Trap Viewer table
🌐	N/A	Shows that you are working with an EMS to manage multiple sites
✓ admin	Shows you if the Web UI is connected to the Config Manager	Shows you if the Web UI is connected to the Config Manager
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	Enables you to log out from the user interface

Tab	**SDC Description**	🌐 **EMS Description**
Switch User	Enables you to access the user interface with a different user name	Enables you to access the user interface with a different user name
API	Enables you to view all the available API methods that can be used in a script	Enables you to view all the available API methods that can be used in a script
Help	Enables you to access the SDC Web UI HTML Help	Enables you to access the EMS Web UI HTML Help

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 a list of statistics graphs.	Displays current system KPI's, statistics graphs and recently generated SNMP traps.
Reports	Disabled	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.
Routing	Provides contextual routing editing interface	Provides contextual routing editing interface (when EMS is installed Routing is globally configured).
Monitoring	Provides an SNMP trap view	Provides real-time view of services and resources in your system
Administration	Provides an interface for administrative procedures such as	Provides an interface for administrative procedures such as

Tab	SDC Description	EMS Description
	user management, backup and restore, etc.	user management, backup and restore, etc.

2.3.3 The Navigation Pane

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

2.3.3.1 The Navigation Pane Filter

As the navigation pane may display many items, you can filter the displayed items according to their name, instead of manually navigating to the destination item.

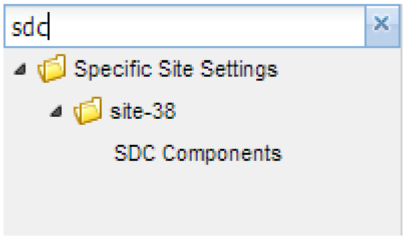
To filter the navigation pane's display:

1. From the Filter field, from the top section of the navigation pane, enter the full name, or the part of the name of the item you are looking for.

   The navigation pane displays the matching results with their path to the root directory that they belong to.

   Figure 2 depicts the displayed results filtered according to the word "sdc":

Figure 2: The Navigation Pane Filter

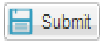
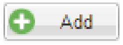
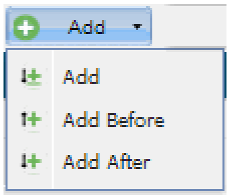
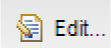
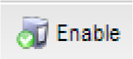
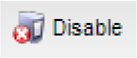
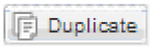
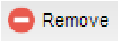
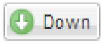
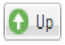
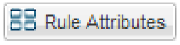
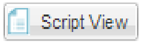
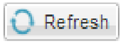
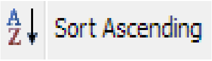
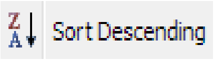
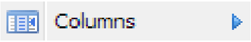


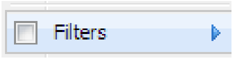
sdc

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
	Saves changes applied to a selected item.
	Adds an item.
	Adds an item in a specific position in the table, relative to the selected item.
	Edits the selected item.
	Sets the **Administrative State** to Enabled for the selected item.
	Sets the **Administrative State** to Disabled for the selected item
	Creates another item in the table with all the definitions of the selected item.
	Removes the selected item.
	Moves the selected item to a lower place in the list.
	Moves the selected item to a higher place in the list.
	Defines the attributes (AVP's) of the rule table.
	Displays the selected rule in script language.
	Refreshes the selected item's properties, 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.

Button	Description
	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.

Note: The keyboard navigation functionality must be enabled during the installation procedure. For more information on the installation procedure, see the *F5 SDC Installation Guide*.

In the current release, keyboard navigation is supported only in Mozilla Firefox.

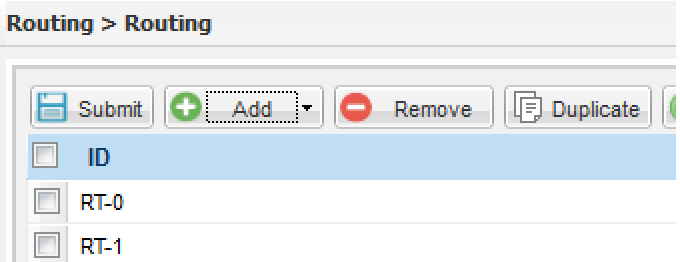
The navigation keys and their corresponding actions are detailed in *Table 5*.

Table 5: Keyboard Navigation

Key	Action
Alt+Ctrl+x	Hotkey to focus on the keyboard button.
Tab	Moving forward to the next element or section
Shift + Tab	Moving backwards to the previous element or section
Arrow Down	Opening a drop-down list
Arrow Up/Arrow Down	Navigation between drop-down list items
Enter	1. Selecting an element from a drop-down list 2. Selecting the tab in focus 3. Accessing a link 4. Opening the accessibility menu when the focus is on the keyboard button

The element in focus is surrounded by a border, as shown in *Figure 3*.

Figure 3: An Element in Focus



□	ID
□	RT-0
□	RT-1

2.3.6 SDC Decision Tables

The SDC Web UI contains the following decision tables:

- Transformation (pre and post)
- Session Management
- Dynamic Peer profile
- Routing

Decision tables are tables of rules, defined by the user, that define how a message is processed at that specific point in the SDC. 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 column with their 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 the rule criteria.

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 rule attribute values defined for the rule that appears first 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 next rule in the decision table is checked, until a rule is found with all the matching criteria.

Configuring the decision tables includes the following procedures:

- *Adding Rule Attributes*
- *Defining Rule Attribute Values*
- *Defining Rule Actions and Configurations*

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

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 **Filter Type** column, from the drop-down list, select the way that the message property is checked against the value defined for this attribute.
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.
8. Click **Submit**. The decision table is now updated with columns reflecting the label values of the added rule attributes.

2.3.6.2 Defining Rule Attribute Values

The Rule Attribute values defined for the decision table Rule Attributes 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*).
   b. No value (leave field empty) – the message and entity will not be checked to see if they contain the property. The rule attribute will automatically be approved and the SDC will move on to check the next rule attribute defined for the rule.

c. NULL – the message and entity will be checked to see if they contain the property. This rule attribute will only be approved if the property does not appear in the message and entity.

Note: When configuring a Rule Attribute for a Routing Profile, only numeric “low-high” value pairs (i.e. 11-22) are supported as a STRING RANGE Filter Type. You cannot apply the STRING RANGE Filter Type to a string-numeric value (i.e. Okano-20).

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.

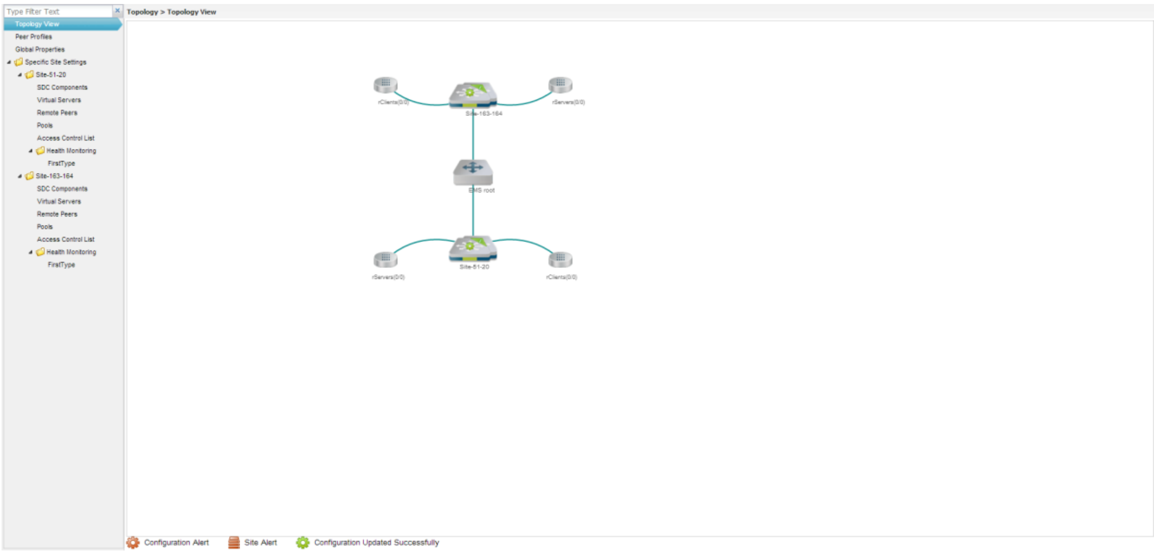
3. Configuring the SDC Topology

This chapter describes how you configure and view the SDC topology, encompassing the different network entities of an SDC deployment. The SDC topology is based on the topology file that was configured during the installation process.

3.1 Topology View

The Topology View provides a bird-eye view over all SDC sites connected to the EMS.

Figure 4: Topology View



3.1.1 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.

3.1.2 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.

To add a global property:

1. Go to **Topology** > **Global Properties** > **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 file path to the property definition is displayed.

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

5. 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*.

3.1.3 Specific Site Settings

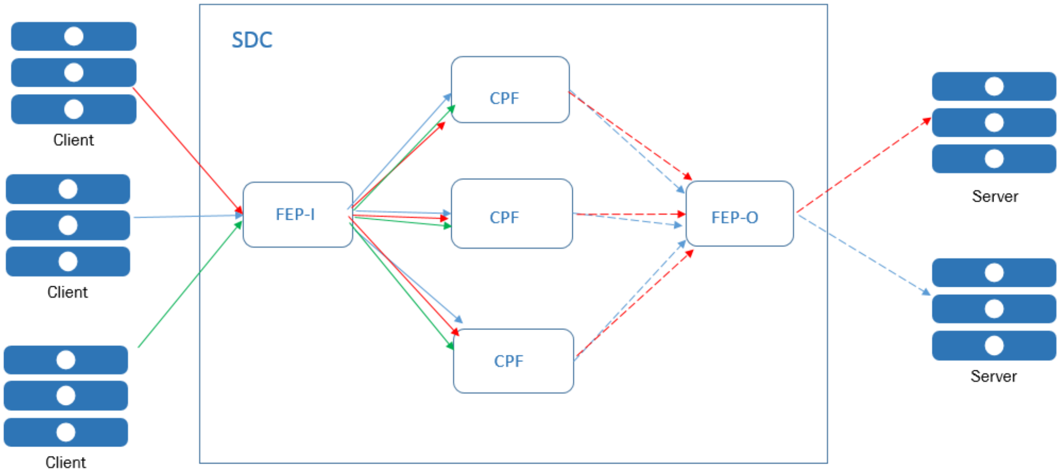
This section describes the different components that are configured per site.

3.1.3.1 SDC Components

SDC comprise 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 connect to all CPF nodes.

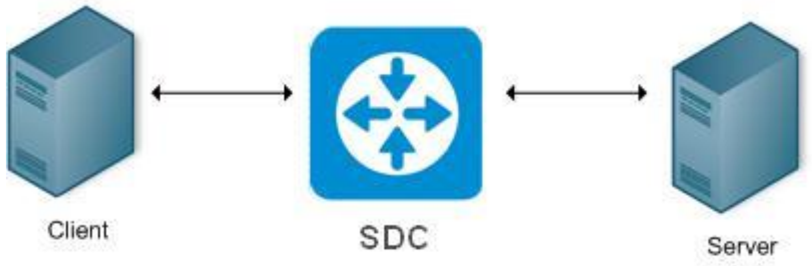
Figure 5 shows the basic network architecture:

Figure 5: Network Architecture



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

Figure 6: 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

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."

Configuring the SDC Components.

3.1.3.2 Virtual Servers

Virtual Servers are virtual instances of SDC used to facilitate every protocol used by SDC to communicate with the Remote Peers (Clients and Servers). 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*.

3.1.3.3 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 Remote Peers*.

3.1.3.4 Pools

Pools are groups of Server Peers. Server 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 Server 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 Server 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 Server 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*.

3.1.3.5 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 host name. 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*.

3.1.4 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.

3.1.5 Topology Architecture

The following section describes the SDC Topology architecture.

Figure 7: SDC Network Topology

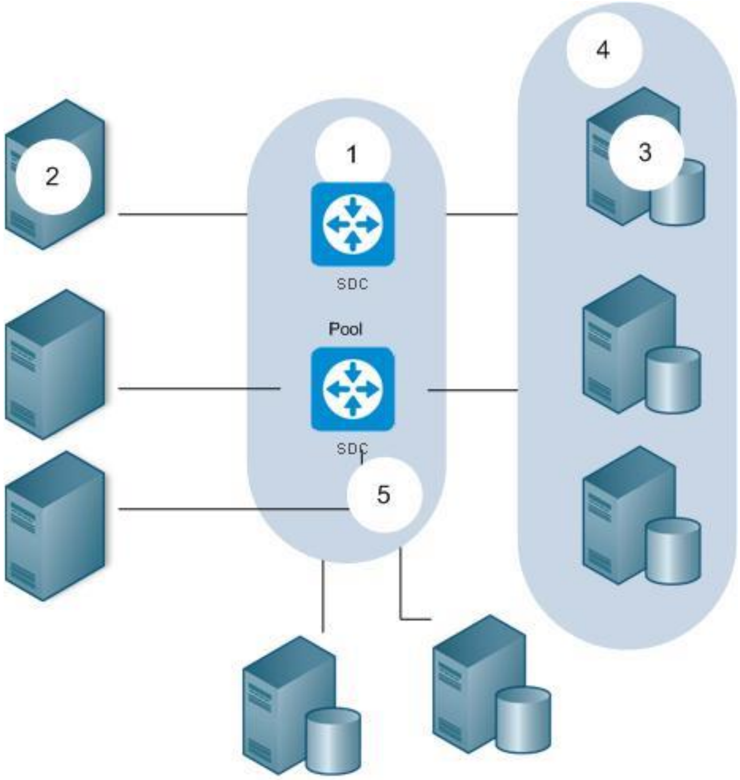


Table 6: SDC Network Topology Legend

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 Server Peers.
5.	Cluster	A group of SDCs used to provide translation and connectivity services and support high availability.

3.2 Configuring the Topology

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

3.2.1 Configuring Peer Profiles

Peer Profiles are logical objects used to tag Remote Peers. Peer Profiles may be assigned Association Rules with which the Remote Peers are compared. When an unknown Remote 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.

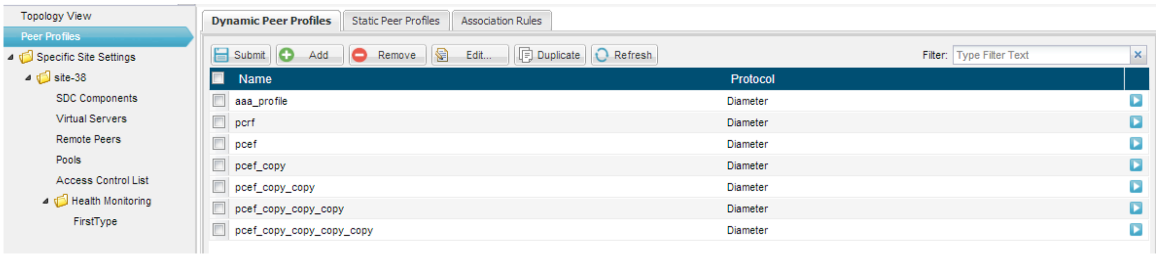
3.2.1.1 Viewing the List of Peer Profiles

You can view the list of available peer profiles.

To view the list of Peer Profiles:

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

Figure 8: Peer Profiles



Name	Protocol
aaa_profile	Diameter
pcrf	Diameter
pcef	Diameter
pcef_copy	Diameter
pcef_copy_copy	Diameter
pcef_copy_copy_copy	Diameter
pcef_copy_copy_copy_copy	Diameter

Table 7 presents a list of peer properties:

Table 7: Peer Profile's Properties

Column	Description
Name	A user friendly display name assigned to the Peer Profile. e.g. PeerProfile1

Column	Description
Protocol	The signaling protocol used by the Remote Peer. e.g. Diameter

3.2.1.2 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:

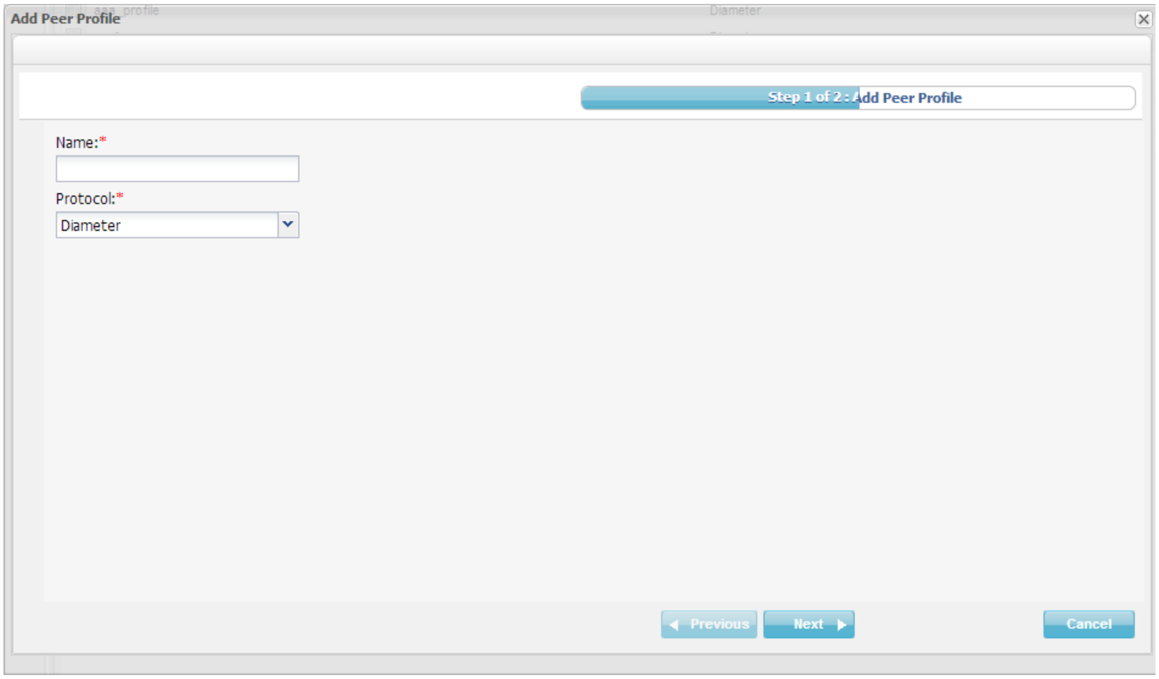
1. Go to **Topology** > **Peer Profiles** and then click **Dynamic Peer Profiles** to create a (client) dynamic Peer Profile or **Static Peer Profiles** to create a static (client or server) Peer Profile.

Note: Server peers can connect dynamically, and be set as servers using a peer profile property.

Predefined static peers (clients or servers) may be applied with a Peer Profile in advance – static or dynamic.

2. Click **Add**. The Add Peer Profile wizard is displayed:

Figure 9: Add Peer Profile Wizard



Add Peer Profile

Step 1 of 2 : Add Peer Profile

Name:*

Protocol:* Diameter

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

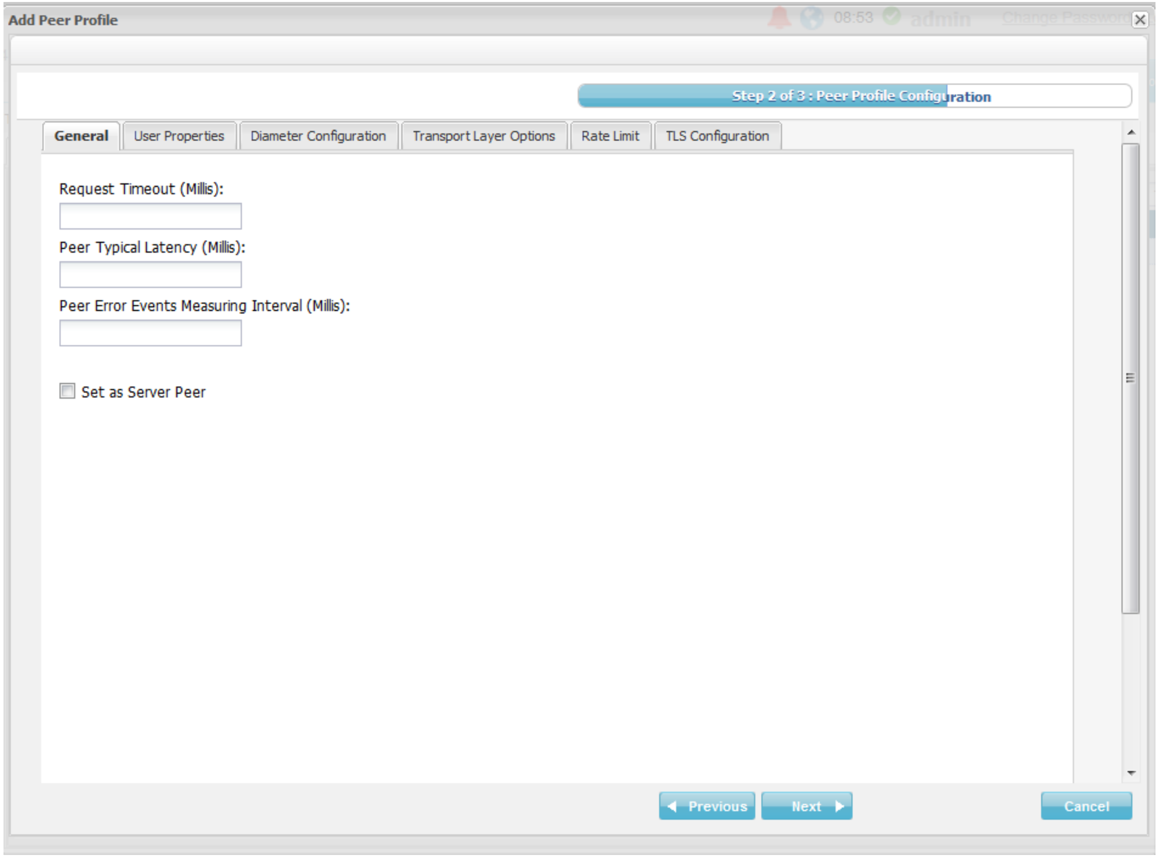
4. In the **Protocol** field, select the signaling protocol used by the Remote Peer, e.g. Diameter.

Note: The SIP protocol is currently not supported.

5. Click **Next**. The Peer Profile Configuration page is displayed:

6. Under **General** tab (available to all protocols):

Figure 10: Peer Profile Configuration



a. In **Request Timeout**, set the time frame in which the Peer is expected to reply requests. Timed-out requests are counted for determining a Server Peer's health. For additional information on Health Monitoring, see *Health Monitoring*.

b. In **Peer Typical Latency (Millis)**, set the typical peer latency time frame.

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

d. Select **Set as Server Peer** if you want to set the unknown Remote Peer as a Server Peer.

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

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**.

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

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.

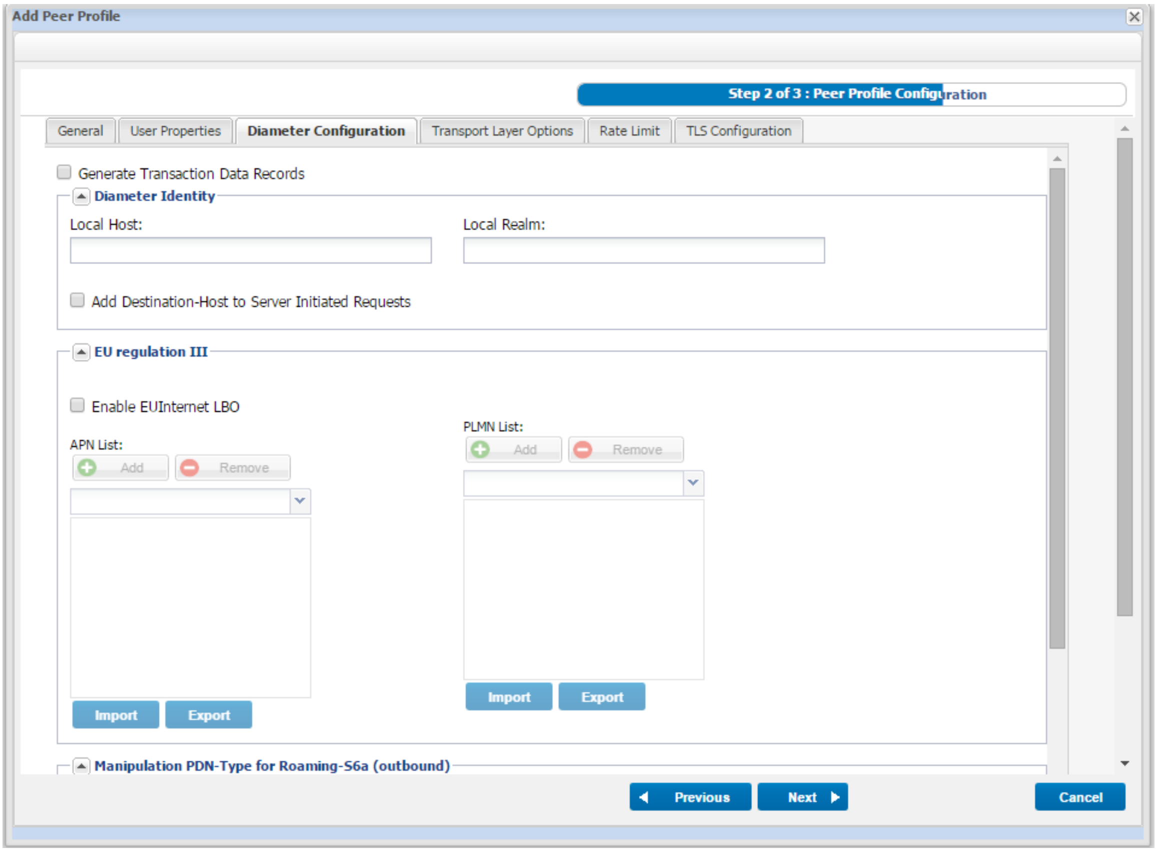
User properties can also be defined using the setEntityProperties Web Service API method and retrieved using the getEntityProperties Web Service API method or using Groovy scripting. For more information about the Web Service API methods, see the *F5 SDC Web Services API Guide*.

3.2.1.2.1 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, and IPv6 - IPv4 Enablement

Figure 11: Diameter Configuration



- ## Diameter Identity

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 *Defining Diameter Identity Parameters*.

To define the Diameter Identity values:

1. In **Local Host**, set the value you wish to appear as the message's origin-host.
2. In **Local Realm**, set the value you wish to appear as the message's origin-realm.
3. 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.

- **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 if 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. Once 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 a connection (Local Breakout) to be established with a VPLMN.

To enable and configure EU Local Breakout:

1. 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.

- **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.

To enable IPv6 protocol for roaming:

1. Select **Enable Manipulation PDN-Type for Roaming S6A (outbound)** for Diameter peer profiles.
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.
3. In the **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.
4. Click **Submit**.

To configure the Transport Layer Options:

1. Under the **Transport Layer Options** tab:
2. Set the parameters that control the behavior of transport layer channels. For information on the transport layer options, see *Default Transport Configuration*.

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*.

To configure the TLS Configuration:

1. Under the **TLS Configuration** tab, select one of the following:
   - **No TLS Security**
   - **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.

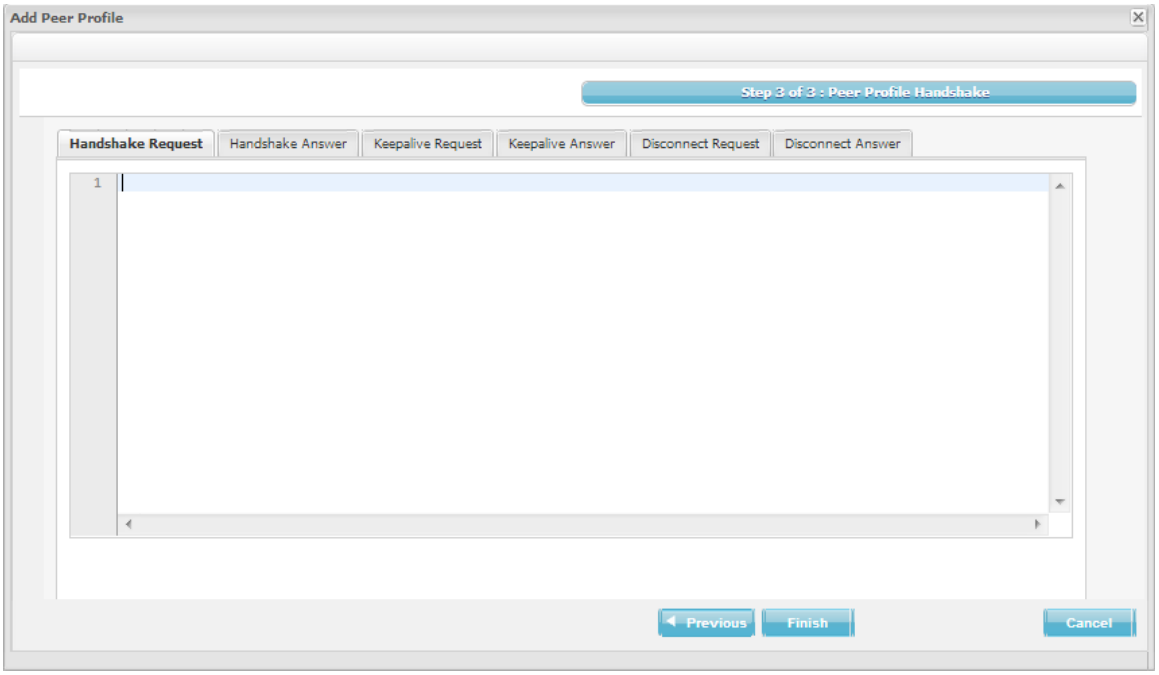
2. If you select either **Pre** or **Post Capabilities Exchange TLS**, you have the option to change the default **TLS Keystore Password** and **TLS Trust Store Password**(s). Default passwords are generated as part of the automatic TLS security key generation. The TLS security key secures the connections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

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

4. Click **Next**. The Peer Profile Handshake page is displayed.

Figure 12: Peer Profile Handshake



5. Under each tab, type in the corresponding script.

Table 8 details the parameters SDC provides to the scripts:

Table 8: Request and/or Answer Scripts Parameters

Parameter	Type
Request	Message
Peer	Peer
Stack	Stack
userTraceLogger	UserTraceLoggerWrapper
metaData	MetaData

3.2.1.2.2 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.

To configure the Transport Layer Options:

2. Under the **Transport Layer Options** tab:

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

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*.

To configure the TLS Configuration:

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

   - **No TLS Security**
   - **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.

4. 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**(s).

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.

5. 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.

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

6. Click **Next**. The Peer Profile Handshake page is displayed.

3.2.1.2.3 LDAP Peer Profile

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

To configure the Transport Layer Options:

7. Under the **Transport Layer Options** tab:
8. Set the parameters that control the behavior of transport layer channels. For information on the transport layer options, see *Default Transport Configuration*.

To configure the rate limit:

1. Under the **Rate Limit** tab (for Diameter, HTTP, Ldap, and File protocols):
2. 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:

1. Under the **TLS Configuration** tab, select one of the following:
   - **No TLS Security**
   - **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**(s). 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. 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.

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

---

4. Click **Next**. The Peer Profile Handshake page is displayed.

3.2.1.2.4 File Peer Profile

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

To configure the Transport Layer Options:

1. Under the **Transport Layer Options** tab:
2. Set the parameters that control the behavior of transport layer channels. For information on the transport layer options, see *Default Transport Configuration*.

To configure the rate limit:

1. Under the **Rate Limit** tab (for Diameter, HTTP, Ldap, and File protocols):
2. 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:

1. Under the **TLS Configuration** tab, select one of the following:
   - **No TLS Security**
   - **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**(s). 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. 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.

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

4. Click **Next**. The Peer Profile Handshake page is displayed.

3.2.1.2.5 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

   i. In **Message-Authenticator Algorithm**, enter the algorithm to be used to authenticate Radius messages.

Note: Messages containing the "EAP-Message" attribute are 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.

    d. In **Retransmission Interval**, set the interval for resending attempts.

To configure the rate limit:

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

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

3.2.1.2.6 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 parameter must be configured to true during installation (or an upgrade) (Configure Properties) 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.

- **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 parameter must be configured to true during installation (or an upgrade) (Configure Properties) 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.

3. In the **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.
4. Click **Finish**.

3.2.2 Association Rules

Association rules are sets of rules according to which Peers are associated with specific Peer Profiles. Each association rule within the table contains a set of parameters, corresponding with a message's content. That is, SDC determines whether or not to associate a Peer with a Peer Profile based on the contents of messages retrieved from it. The messages' parameters are represented by Association Rule Attributes – AVPs. The Association Rule attributes are configured independently and each AVP is assigned a type (Boolean, regular expression, etc.).

The Rule Attributes list may, for example, consist of the AVP OriginHost. When setting the association rules you may use this AVP to determine certain origin host Peers that are associated with this Peer Profile.

Association rules are scanned in the order they are listed. The first association rule's condition that is met (that is, the message's attributes match the rule's criteria) associates its selected Peer Profile with the Peer from which the message was retrieved.

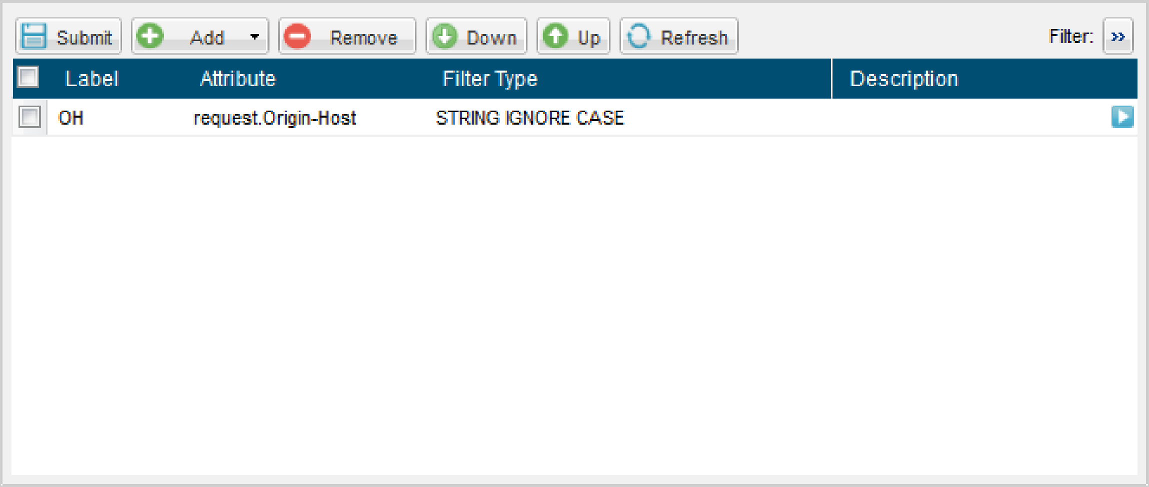
The **Association Rules** tab displays the currently empty list of routing rules. To define the routing rules you first need to define their attributes. This is configured in the **Association Rule Attributes** tab.

3.2.2.1 Adding a New Association Rule Attribute

To add an Association Rule attribute:

1. Click **Rule Attributes**. The tab displays the list of attributes (AVPs) that may be used to define the routing rules:

Figure 13: Association Rule Attributes



□	Label	Attribute	Filter Type	Description
□	OH	request.Origin-Host	STRING IGNORE CASE	

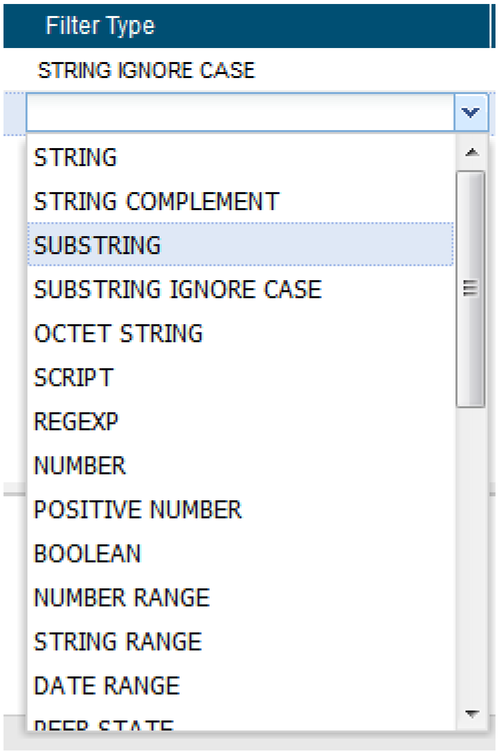
2. Click **Add**. A new line is added to the table.

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 **Filter Type**, select the data type of the new attribute. e.g.: String

Figure 14: Attribute Types



6. Under **Description**, type in a short description of the attribute.

7. Click **Submit**.

Note: For additional information on the decision table attributes, see *Appendix D: Decision Table Attributes* .

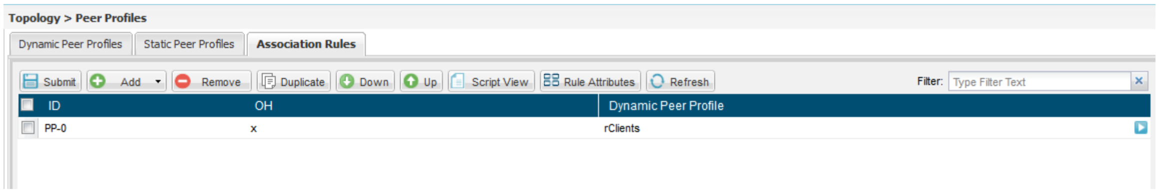
3.2.2.2 The Association Rules

This section describes how to add association rules.

To add a new association rule:

1. Click the **Association Rules** tab. The tab displays a table. The table’s columns represent the Association Rule Attributes that you previously defined.

Figure 15: Association Rules



☐	ID	OH	Dynamic Peer Profile
☐	PP-0	x	rClients

2. Click **Add** to create a new association rule. A new rule line assigned an automatic name is added to the table.

3. Under each column, select the value against which messages are compared. This rule shall associate the selected Peer Profile with the Peer from which matching messages are retrieved.

4. Under **Dynamic Peer Profile**, from the previously created list of Dynamic Peer Profiles, select the Peer Profile to associate with the unknown matching Peer.

3.2.3 Configuring a Site's User Properties

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:

5. Go to **Topology** > **Site** > **Add**.

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

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

8. 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."

3.2.4 Configuring the SDC Components

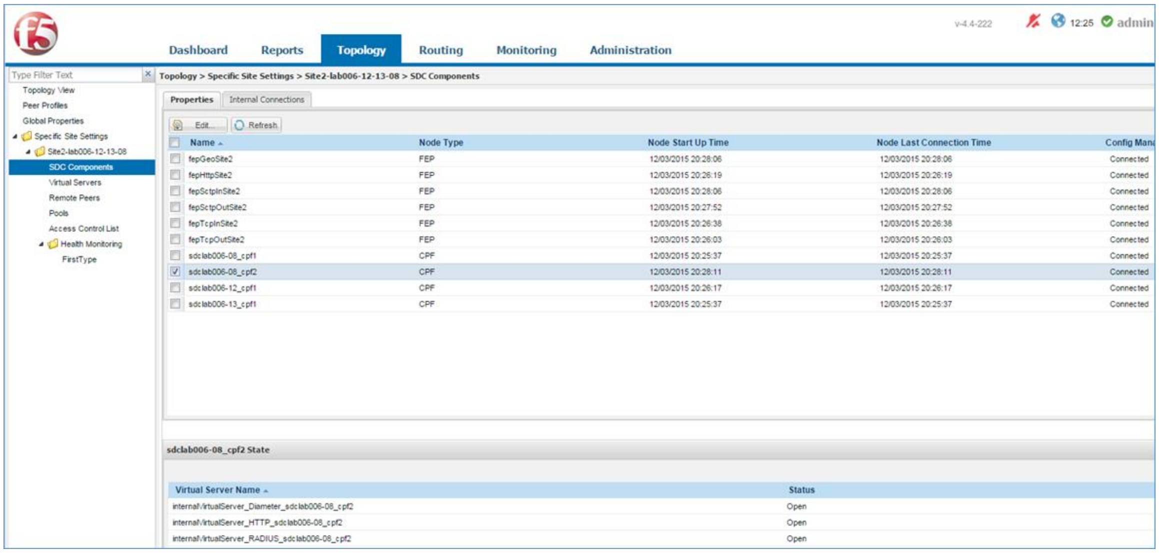
This section describes how to view and edit the SDC Components that were defined during the installation process.

To edit the SDC Component list:

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

   The list of SDC Components defined throughout the installation procedure is displayed according to the properties described in *Table 9*.

Figure 16: SDC Components



Topology > Specific Site Settings > Site2-lab006-12-13-08 > SDC Components

□	Name	Node Type	Node Start Up Time	Node Last Connection Time	Config Mana
□	fepGeoSite2	FEP	12/03/2015 20:28:06	12/03/2015 20:28:06	Connected
□	fepHttpSite2	FEP	12/03/2015 20:26:19	12/03/2015 20:26:19	Connected
□	fepSctpInSite2	FEP	12/03/2015 20:28:06	12/03/2015 20:28:06	Connected
□	fepSctpOutSite2	FEP	12/03/2015 20:27:52	12/03/2015 20:27:52	Connected
□	fepTcpInSite2	FEP	12/03/2015 20:26:38	12/03/2015 20:26:38	Connected
□	fepTcpOutSite2	FEP	12/03/2015 20:26:03	12/03/2015 20:26:03	Connected
□	sdclab006-08_cpf1	CPF	12/03/2015 20:25:37	12/03/2015 20:25:37	Connected
☑	sdclab006-08_cpf2	CPF	12/03/2015 20:28:11	12/03/2015 20:28:11	Connected
□	sdclab006-12_cpf1	CPF	12/03/2015 20:26:17	12/03/2015 20:26:17	Connected
□	sdclab006-13_cpf1	CPF	12/03/2015 20:25:37	12/03/2015 20:25:37	Connected

sdclab006-08_cpf2 State

Virtual Server Name	Status
internalVirtualServer_Diameter_sdclab006-08_cpf2	Open
internalVirtualServer_HTTP_sdclab006-08_cpf2	Open
internalVirtualServer_RADIUS_sdclab006-08_cpf2	Open

Table 9: SDC Components

Column	Description
Name	A user friendly name (derived from the Node's URI).
Node Type	CPF (Control Plane Function) or FEP (Front End Proxy). FEP is a network distribution point in an F5 application. It is built on top of the CPF framework to take advantage of the CPF management, pipeline and other infrastructures. FEP maintains a steady single connection of TCP with the multiple CPF nodes. For each Remote Node,

Column	Description
	it manages the connection and state machine, providing statistics and management capabilities for the connections and the traffic. FEP is responsible for the following functionalities: Connection maintenance Licensing ACL Message security Peer Profile ACL Peer State machine Flow control CPF is responsible for: Flow control Session management Routing management Transformation Tracing Data Transaction
Node Start Up Time	Indicated the node’s startup time
Node Last Connection Time	Indicated the last time in which the node was connected
Config Manager Connection State	Indicates the current connection state (**Connected**/**Disconnected**)

2. Select each SDC Component to show the list of its **Virtual Server Name**(s) and their **Status** as displayed in the bottom pane of the SDC Components screen.

To edit an SDC Component's properties:

1. Select a row of a specific SDC Component and click **Edit**.

   The SDC Component Properties window displays the following properties:

   - **General**
   - **Diameter**
   - **SS7**
   - **User Properties**

2. After editing the parameter properties, click **Submit**.

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 *F5 SDC Installation Guide*.
Scripts Update Time	The last date SDC has received a script update from the Management Console. If this date does not match the last date a configuration update has occurred, contact F5 Technical Support.
TCTimer (Millis)	The interval for reconnecting the SDC component.
Product Name	The product name of the SDC Component, published during capability exchange.
Configuration Update Time	The last date SDC has received a configuration update from the Management Console, or the last date when SDC has started-up.

Table 11: Diameter SDC Component's Properties

Parameter	Description
TWTimer (Millis)	Watchdog and reconnection timer (in Milliseconds). e.g. 30000. Note: The minimum TWTimer value is 6000 milliseconds.
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* .
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

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

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 VIPs*.

3.2.4.1 Viewing the Internal Connections

Each FEP node is connected to each of the CPF Nodes in SDC and vice versa. The Internal Connections table displays the entire list of FEP-CPF connections, in which the FEP node serves as a client and the CPF node serves as a server.

Table 14 details the Internal Connections' table.

Table 14: Internal Connections

Column	Description
Client Name	The FEP Node
Client Address	The FEP Node's address
Server Name	The CPF Node

Column	Description
Server Address	The CPF Node's address
Status	The connection status between the FEP and the CPF nodes

3.2.5 Configuring Virtual Servers

Virtual Servers are virtual instances of SDC used to facilitate every protocol used by SDC to communicate with the Remote Clients and Remote Servers. 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.

3.2.5.1 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:

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

   The list of Virtual Servers is displayed according to the properties described in *Table 15*.

Table 15: Virtual Server Properties

Column	Description
Name	A user friendly display name assigned to the Virtual Server. e.g. VS1
Addresses	The address (single or multiple) of the Virtual Server and The port number used by it.
Proxy 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

Column	Description
Status	Indicates if the Virtual Server is available (✓) or not available (✖) to receive traffic.

To edit a Virtual Server Property:

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

3.2.5.2 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.

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

Note: The SIP protocol is not supported in this release.

4. If you selected **Diameter**, **RADIUS**, or **HTTP**, and you want to set the virtual server on a specific Proxy (FEP Node), select **Use Proxy**.
5. Click **Next**. The wizard proceeds to the next step according to your protocol selection. Proceed to the next selected 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 an open status (✓) by default, even before it is licensed. However, if no license has been assigned to the newly added virtual server, messages to the virtual server will not be able to be processed and the following WARN message will appear in the FEP log:

“System Client [client name] was rejected because CPF is not licensed to listen on IP: [IP]. Check the license file and verify that your IP Address is included in the file.” If you receive this message, then you need to configure a new license for the added virtual server. For more information about adding a license, see *Licensing the VIPs.*

3.2.5.2.1 Diameter Virtual Server

This section continues with step 2 of the Add Virtual Server wizard for adding a Diameter virtual server.

To add a Diameter virtual server:

1. In **Proxy Group**, select the Proxy Node (FEP Node) on which the virtual server is set.

Note: This field is only displayed when selecting **Use Proxy** in the previous wizard step.

2. In **Addresse**s, set the address (single or multiple) of the Virtual Server.
3. In **Port**, enter the port on which the virtual server is listening.
4. In **Peer Profile**, select the Peer Profile associated with this Virtual Server.
5. Select **Use SCTP Transport** to use SCTP when in message transport (rather than TCP).

6. Click **Finish**. The new Diameter Virtual Server is displayed in the Virtual Server table.

3.2.5.2.2 RADIUS Virtual Server

This section continues with step 2 of the wizard for adding a RADIUS virtual server.

To add a Radius virtual server:

1. In **Proxy Group**, select the Proxy Node (FEP Node) on which the virtual server is set.

> Note: This field is only displayed when selecting **Use Proxy** in the previous wizard step.

2. In **Addresse**s, set the address (single or multiple) of the Virtual Server.
3. In **Port**, enter the port on which the virtual server is listening.
4. In **Peer Profile**, select the Peer Profile associated with this Virtual Server.
5. Click **Finish**. The new RADIUS Virtual Server is displayed in the Virtual Server table.

3.2.5.2.3 HTTP Virtual Server

This section continues with step 2 of the wizard for adding a HTTP virtual server.

To add an HTTP virtual server:

1. In **Proxy Group**, select the Proxy Node (FEP Node) on which the virtual server is set.

> Note: This field is only displayed when selecting **Use Proxy** in the previous wizard step.

2. In **Addresse**s, set the address (single or multiple) of the Virtual Server.
3. In **Port**, enter the port on which the virtual server is listening.

4. In **Peer Profile**, select the Peer Profile associated with this Virtual Server.
5. Select **Close Connection on Answer** to close the connection with the Remote Client/Server upon Answer retrieval.
6. Click **Finish**. The new RADIUS Virtual Server is displayed in the Virtual Server table.

3.2.5.2.4 LDAP Virtual Server

This section continues with step 2 of the wizard for adding an LDAP virtual server.

To add an LDAP virtual server:

1. In **Addresses**, set the address (single or multiple) of the Virtual Server.
2. In **Port**, enter the port on which the virtual server is listening.
3. In **Num Acceptor Threads**, set number of threads to be used.
4. In **Back Log**, type in the queue size for incoming LDAP messages waiting to be handled by the LDAP virtual server.
5. Select **Bind User** to mandate user credentials.
   a. In **Bind User**, type in the LDAP user for the directory server authentication.
   b. In **Bind Password**, type in the LDAP user's password for the directory server authentication.
6. Select **Anonymous Bind** to allow users to connect to the directory without user credentials.
7. Click **Finish**. The new LDAP Virtual Server is displayed in the Virtual Server list.

3.2.6 Configuring Remote Peers

This section describes how to configure the different remote peers,

3.2.6.1 Viewing the List of Remote Peers

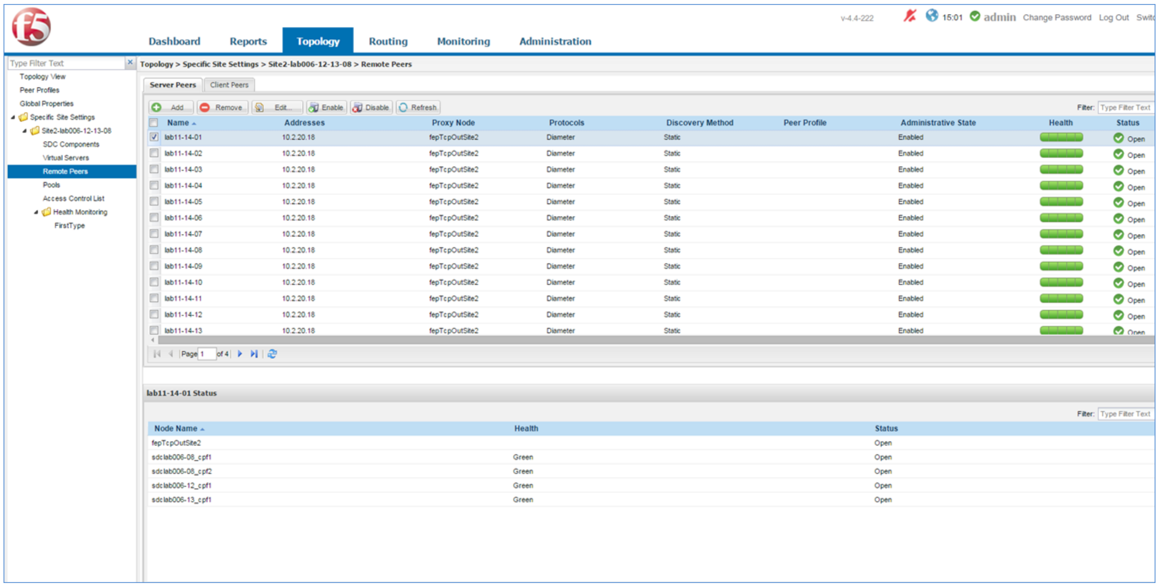
To view the current list of Remote Peers:

1. Go to **Topology** > **Specific Site Settings** > **Site** > **Remote Peers**.

   The list of currently defined Remote Peers is displayed, divided into two tabs: **Server Peers** and **Client Peers**:

Note: To disable auto refresh of the screen data, switch the **Auto Refresh** button from ON to OFF. Auto-refresh is enabled by default, and must be disabled every time the screen is accessed.

Figure 17: Remote Peers



Topology > Specific Site Settings > Site2-lab006-12-13-08 > Remote Peers

☐	Name	Addresses	Proxy Node	Protocols	Discovery Method	Peer Profile	Administrative State	Health	Status
☑	lab11-14-01	10.2.20.18	fepTcpOutSite2	Diameter	Static		Enabled	🟩	✓ Open
☐	lab11-14-02	10.2.20.18	fepTcpOutSite2	Diameter	Static		Enabled	🟩	✓ Open
☐	lab11-14-03	10.2.20.18	fepTcpOutSite2	Diameter	Static		Enabled	🟩	✓ Open
☐	lab11-14-04	10.2.20.18	fepTcpOutSite2	Diameter	Static		Enabled	🟩	✓ Open
☐	lab11-14-05	10.2.20.18	fepTcpOutSite2	Diameter	Static		Enabled	🟩	✓ Open
☐	lab11-14-06	10.2.20.18	fepTcpOutSite2	Diameter	Static		Enabled	🟩	✓ Open
☐	lab11-14-07	10.2.20.18	fepTcpOutSite2	Diameter	Static		Enabled	🟩	✓ Open
☐	lab11-14-08	10.2.20.18	fepTcpOutSite2	Diameter	Static		Enabled	🟩	✓ Open
☐	lab11-14-09	10.2.20.18	fepTcpOutSite2	Diameter	Static		Enabled	🟩	✓ Open
☐	lab11-14-10	10.2.20.18	fepTcpOutSite2	Diameter	Static		Enabled	🟩	✓ Open
☐	lab11-14-11	10.2.20.18	fepTcpOutSite2	Diameter	Static		Enabled	🟩	✓ Open
☐	lab11-14-12	10.2.20.18	fepTcpOutSite2	Diameter	Static		Enabled	🟩	✓ Open
☐	lab11-14-13	10.2.20.18	fepTcpOutSite2	Diameter	Static		Enabled	🟩	✓ Open

lab11-14-01 Status

Node Name	Health	Status
fepTcpOutSite2		Open
sdclab006-08_cpf1	Green	Open
sdclab006-08_cpf2	Green	Open
sdclab006-12_cpf1	Green	Open
sdclab006-13_cpf1	Green	Open

Table 16 presents a list of Remote Client/Server Peers.

Table 16: Remote Peer's Properties

Column	Description
Name	A user friendly display name assigned to the Remote Peer. e.g. Server 1

Column	Description
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
Proxy	The name of the FEP Node to which the Remote Peer is connected
Protocols	The signaling protocol/s used by the client or server. e.g. Diameter/JMS
Discovery Method	Specifies whether the Remote 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 Remote Peer. Remote Peers assigned with a Peer Profile are handled in a predefined manner – they may receive unique messages and send unique messages.
Administrative State	Indicates whether the Remote Peer is enabled or not.
Health	Indicates the health status of each remote peer. The health status is calculated based on the peer performance measured periodically and the rate limit predefined by the user. The peer health is presented to the user in the Remote Peers screen with one of three possible icons – (🟩, 🟨, 🟥) – representing the different peer health states. The color of each peer health range is based on ranges of up to 20% (red bar), between 20% and 80% (yellow bar), and between 80% and 100% (green bar). Any “out of service”, “out of service partially” and disconnected peers are marked with a “red bar” health (0%).
Status	Indicates whether the Remote Peer is currently connected to an SDC.

Column	Description
	▪ 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 out of service partially, 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 .
Node Health	Indicates the health (**Green**, **Yellow**, and **Red**) of the peer per CPF.
Node 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.

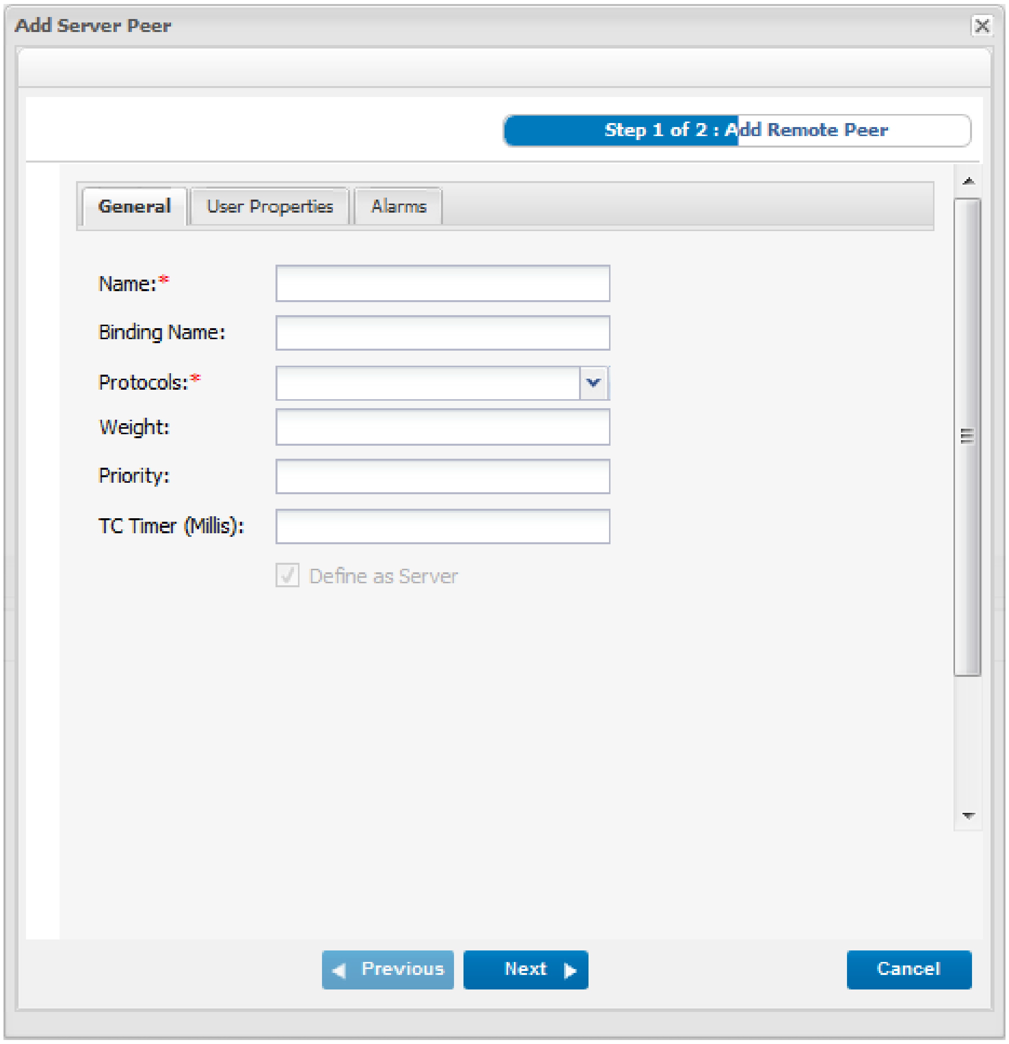
3.2.6.2 Adding a New Remote Peer

This section describes how to add a new remote peer.

To add a new Remote Peer:

1. From the **Server Peer** tab or the **Client Peer** tab, click Add. The Add Server Peer/Add Client Peer window appears:

Figure 18: Add Server Peer Window



Add Server Peer

Step 1 of 2 : Add Remote Peer

Name:*

Binding Name:

Protocols:*

Weight:

Priority:

TC Timer (Millis):

☑ Define as Server

2. In the **General** tab:

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

Note: After submitting the new Remote Peer, its name may not be modified.

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

c. In **Protocols**, out of the available signaling protocols, select the protocol used by the Client Peer or Server Peer from the drop-down list.

Note: The SIP protocol is not supported in this release. If you selected an SS7 protocol, proceed to *SS7 Remote Peer* for the steps on how to configure an SS7 Remote Peer.

d. In **Weight**, set the Remote Peer's weight (a number) in traffic distribution, in case it is included in Weighted Round Robin Pool or in Contextual Pool.

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

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

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

f. In **TC Timer (Millis)**, type in the reconnection attempts interval.

g. If you selected **Diameter** or **RADIUS**, and you want this peer to connect to a FEP node as its proxy node, select **Use Proxy**.

Note: **Define a Server** is non-configurable, and reflects the type of remote peer that is configured.

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**.

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

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*.

4. Under the **Alarms** tab, enter the threshold percentages (**Critical**, **Major**, **Minor)** for the **TPS Rate Limit** for the Remote Peer. For more information about threshold management and how it can be configured globally, see *Threshold Management*.

3.2.6.2.1 Diameter Remote Peer

This section continues with step 2 of the wizard for adding a Diameter remote peer.

To add a server peer with Diameter properties:

1. In **Proxy Group**, select the FEP node, from the drop-down list to which the Remote Peer is connecting.
2. In **Host IP Addresses**, set the address (single or multiple) where the Remote Peer (client or server) is hosted.
3. In the **Port** field, specify an available port number for the Remote Peer to access the SDC Components.
4. In **Local IP Addresses**, type in the IP Address from which to send messages.
5. In **Local Port**, set the local port from which to send messages to a Server Remote Peer.

6. 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*.

---

7. In **TW Timer (Millis)**, set the Watchdog and reconnection timer (in Milliseconds).
8. In **Local Host**, set a value for the message's origin-host.
9. In **Local Realm**, set a value for the message's origin-realm.
10. Select **Use SCTP Transport** to use SCTP (rather than TCP) for message transport.
11. Select **Use for Geo Redundant Sites Connection** to enable the peer to handle replicated data originating from specific sessions.

Note: Only one peer per site can be configured as a site replicator peer (by selecting the **Use for Geo Redundant Sites Connection** checkbox). For additional information on Site Replication, see *Site Replication*.

---

12. Click **Finish**. The new Remote Peer is displayed in the Server Peer table or the Client Peer table, according to your selection.

3.2.6.2.2 RADIUS Remote Peer

This section continues with step 2 of the wizard for adding a RADIUS remote peer.

To add a server peer with RADIUS properties:

1. In **Proxy Group**, select the FEP node, from the drop-down list to which the Remote Peer is connecting.
2. In **Host IP Addresses**, set the address (single or multiple) where the Remote Peer (client or server) is hosted.

3. In the **Port** field, specify an available port number for the Remote Peer to access the SDC Components.
4. In **Local IP Addresses**, type in the IP Address from which to send messages.
5. In **Local Port**, set the local port from which to send messages to a Server Remote Peer.
6. 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*.

7. 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.
8. In **Connection Pool Size**, set the maximum number of open RADIUS connections.
9. Click **Finish**. The new Remote Peer is displayed in the Server Nodes table or the Client Nodes table, according to your selection.

3.2.6.2.3 HTTP Remote Peer

This section continues with step 2 of the wizard for adding a HTTP remote peer

To add a server peer with HTTP properties:

1. In **Host IP Addresses**, set the address (single or multiple) where the Remote Peer (client or server) is hosted.
2. In the **Port** field, specify an available port number for the Remote Peer to access the SDC Components.
3. In **Local IP Addresses**, type in the IP Address from which to send messages.

4. 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*.

5. In **TW Timer (Millis)**, set the Watchdog and reconnection timer (in Milliseconds).
6. In **Max Connection Count Limit (Per Server)**, set the maximum number of open HTTP connections.
7. Select **Keep Alive** to preserve a persistent HTTP connection.
8. Click **Finish**. The new Remote Peer is displayed in the Server Nodes table or the Client Nodes table, according to your selection.

3.2.6.2.4 LDAP Remote Peer

This section continues with step 2 of the wizard for adding a LDAP remote peer.

To add a server peer with LDAP properties:

1. In **Host IP Addresses**, set the address (single or multiple) where the Remote Peer (client or server) is hosted.
2. In the **Port** field, specify an available port number for the Remote Peer to access the SDC Components.
3. 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*.

4. 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 **LDAP Pool Size**, specify the number of connections to use while connecting the LDAP remote peer.
7. Click **Finish**. The new Remote Peer is displayed in the Server Nodes table or the Client Nodes table, according to your selection.

3.2.6.2.5 File Remote Peer

This section continues with step 2 of the wizard for adding a File remote peer.

To add a File server peer:

1. Set Primary **IP** of the File Server.
2. Set **Primary Port** of the File Server.
3. In the **Split By field**, set the value for which the messages will be divided into groups.
4. In the **Number of Groups** field, set how many groups will be needed.
5. In the **FTP Server Name**, select the FTP server for uploading the files from this peer.
6. Click **Finish**. The new Remote Peer is displayed in the Server Nodes table or the Client Nodes table, according to your selection.

3.2.6.2.6 SS7 Remote Peer

This section continues with step 2 of the wizard for adding an SS7 Remote 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.”

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 **Application protocol**, type the protocol of the expected messages.
3. In **Map Version**, type in the SS7 version of the expected messages.
4. Select **Diameter-GSM MAP Auto Transformation Enabled** to automatically transform messages.
5. Select **Route on Global Title** to route messages of this peer using its global title indicator (the SS7 IP equivalent).
    a. In **Encoding Scheme**, type one for odd numbers or two for even numbers.
    b. In **Nature of Address Indicator**, type three for national addresses or four for international addresses.
    c. In **Global Title Address**, type in the peer’s address (maximum length is 15 digits)
    d. In **Global Title Indicator**, type in 4 for Global Title format.
    e. In **Translation Type**, type in 0.
    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 **Finish**. The new Remote Peer is displayed in the Server Nodes table or the Client Nodes table, according to your selection.

3.2.6.2.7 SIP Server Remote PeerEditing a Remote Peer

This section describes how to edit a remote peer.

Note: You may only edit Server Peers.

To edit the Remote Peer:

1. Select the Remote Peer from the Server Peer/Client Peer list and click **Edit**. The Edit Server Peer/Edit Client Peer wizard appears.
2. You may edit the enabled fields, as detailed in *Adding a New Remote Peer*.

Note: **Name**, **Protocol** and **Define as Server** parameters cannot be edited.

3.2.6.3 Removing a Remote Peer

This section describes how to remove any of the remote peers from the list.

To remove a Remote Peer from the list:

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

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

Note: When a Remote 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*.

To refresh the Remote Peer list:

1. Click **Refresh**.

3.2.7 Configuring Pools

Pools are groups containing server 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*
- *Editing a Pool*
- *Removing a Pool*

3.2.7.1 Viewing a List of Pools

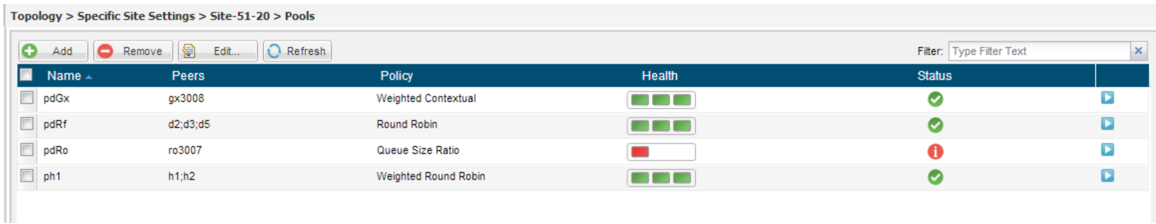
You can view the current list of configured pools.

To view the list of pools:

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

> Note: To disable auto refresh of the screen data, switch the **Auto Refresh** button from ON to OFF. Auto-refresh is enabled by default, and must be disabled every time the screen is accessed.

Figure 19: Pools



Topology > Specific Site Settings > Site-51-20 > Pools

□	Name	Peers	Policy	Health	Status
□	pdGx	gx3008	Weighted Contextual	🟩🟩🟩	✓
□	pdRf	d2;d3;d5	Round Robin	🟩🟩🟩	✓
□	pdRo	ro3007	Queue Size Ratio	🟥	!
□	ph1	h1;h2	Weighted Round Robin	🟩🟩🟩	✓

Table 17 presents a list of Pool’s properties.

Table 17: Pool’s Properties

Column	Description
Name	A user friendly display name assigned to the pool. e.g. Pool1
Peers	The list of Server 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, refer to 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 within the pool. The pool health is calculated as a weighted average of its peers’ health according to their TPS. 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 to the user in the Pools screen with one of three possible icons – – (🟩 , 🟨 , 🟥) – representing the different pool health states. The color of each pool health range is based on ranges of up to 20% (red), between 20% and 80% (yellow), and between 80% and 100% (green).
Status	Indicates the availability of the pool. ▪ 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 indicates as ✓ . ▪ When a pool is open for some CPFs and out of service for other CPFs, the status is indicated as ❗. ▪ When all the CPFs are out of service for the pool, the status is indicated as ✖ .
Node Health	Indicates the health (**Red**, **Yellow**, and **Green**) of the pool per connected CPF.
Node Status	Indicates the status (**Open**, **Out of service**, and **Closed**) of each connected FEP or CPF.

3.2.7.2 Adding a New Pool

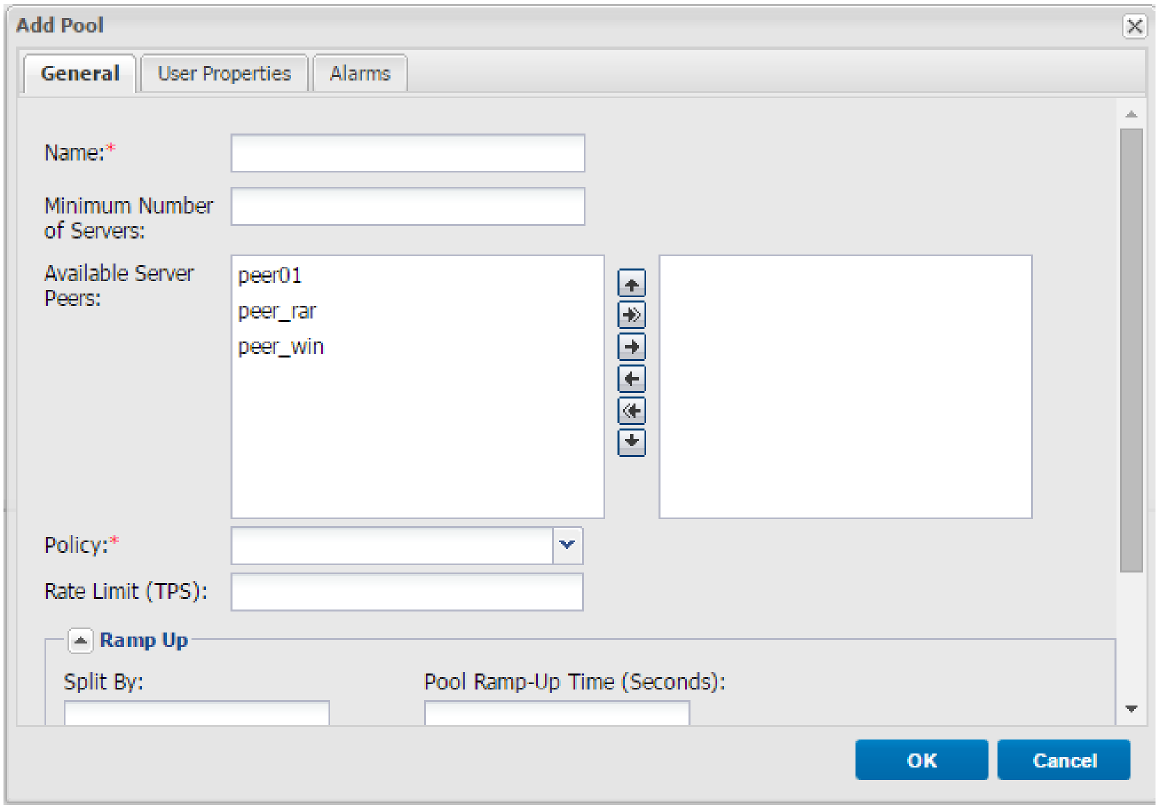
You can add a new pool and define which server peers belong to the added pool.

To add a new pool:

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

The Add Pool dialog box is displayed:

Figure 20: Add Pool



2. In the **General** tab:

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

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

   b. In **Minimum Number of Servers**, enter the number of servers that are available.

The **Minimum Number of Servers** 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 server in the pool is available, an Error event occurs. The default value of Minimum Number of Peers is 1.

c. From the **Available Server Peers** box, click to select the peer(s) that you want to include in the pool.

i. Click the single right arrow button. The Server Peer is added to the pool.

ii. Repeat the above for each peer you want to add to the pool.

iii. To add all available Server Peers to the pool, click the double right arrow button.

iv. To remove a Server Peer from the pool, click to select it from the right box and then click the left arrow button. To remove all Server Peers from the pool click the double left arrow button.

a. From the **Policy** drop-down list, 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*.

b. In **Rate Limit (TPS)**, enter the maximum TPS that can be processed by the pool.

c. In the **Ramp Up** section:

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

ii. 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

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

d. In the **Dynamically add Peers matching the following Peer Profile(s)**, select the relevant peer profiles that you want to apply this option.

Note: This assumes that you have configured the relevant dynamic peer profiles in **Topology > Peer Profiles >Dynamic Peer Profiles**.

e. Click **OK**.

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

Using the User Properties, you can create additional properties for the Remote Peer and define the value for these properties. These properties can be used in scripts relating to the Remote Peer. 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 **OK**.

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 the **Alarms** tab, enter the threshold percentages (**Critical**, **Major**, **Minor)** for the **TPS Rate Limit** for the pool. For more information about threshold management and how it can be configured globally, see *Threshold Management*.

3.2.7.3 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:

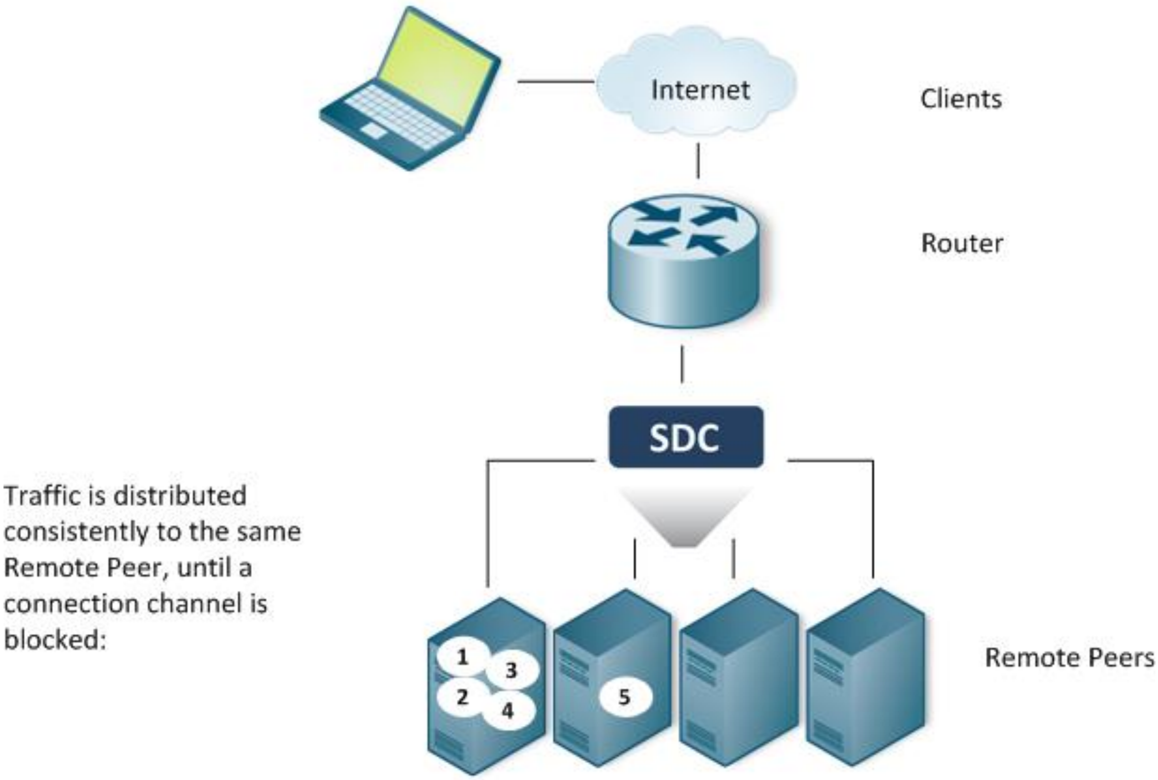
1. Go to **Topology** > **Pools** > **Add/Edit** > **General** > **Policy**.
2. Select the relevant policy from the drop-down list.

3.2.7.3.1 By Precedence

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 21*.

Figure 21: By Precedence Policy


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

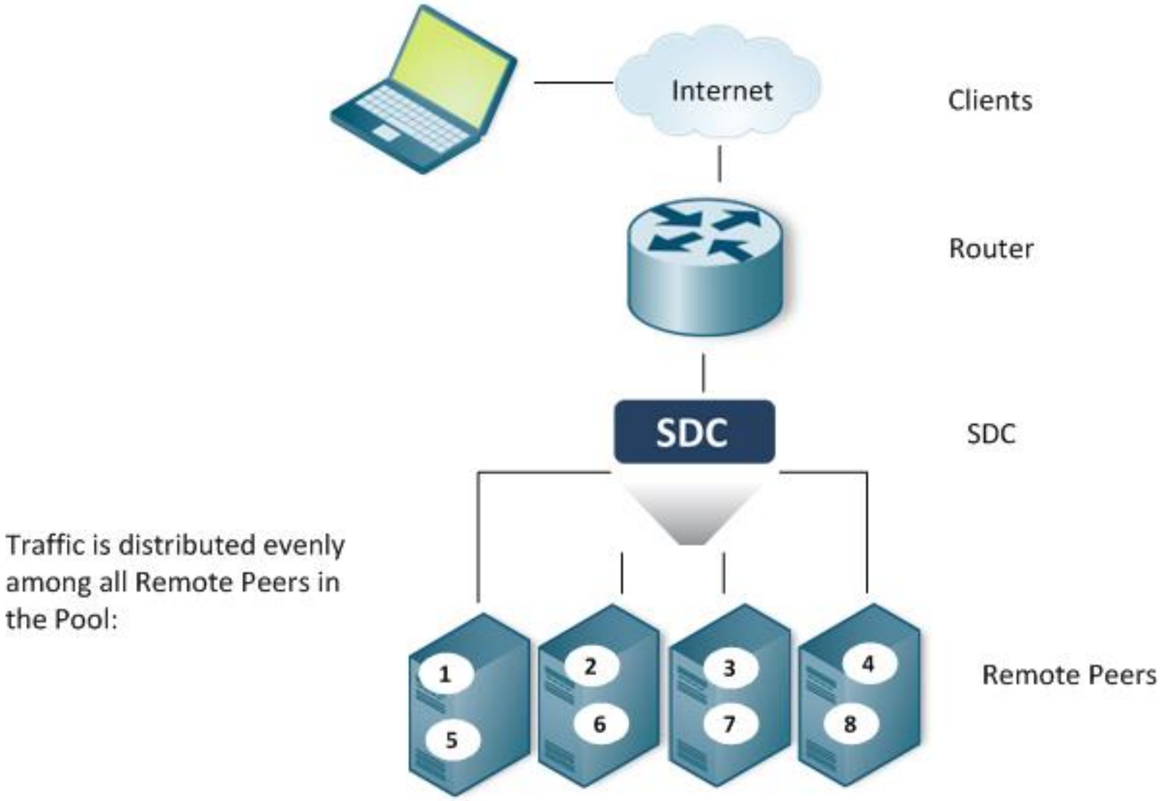
3.2.7.3.2 Round Robin

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 22*.

Figure 22: Round Robin Policy

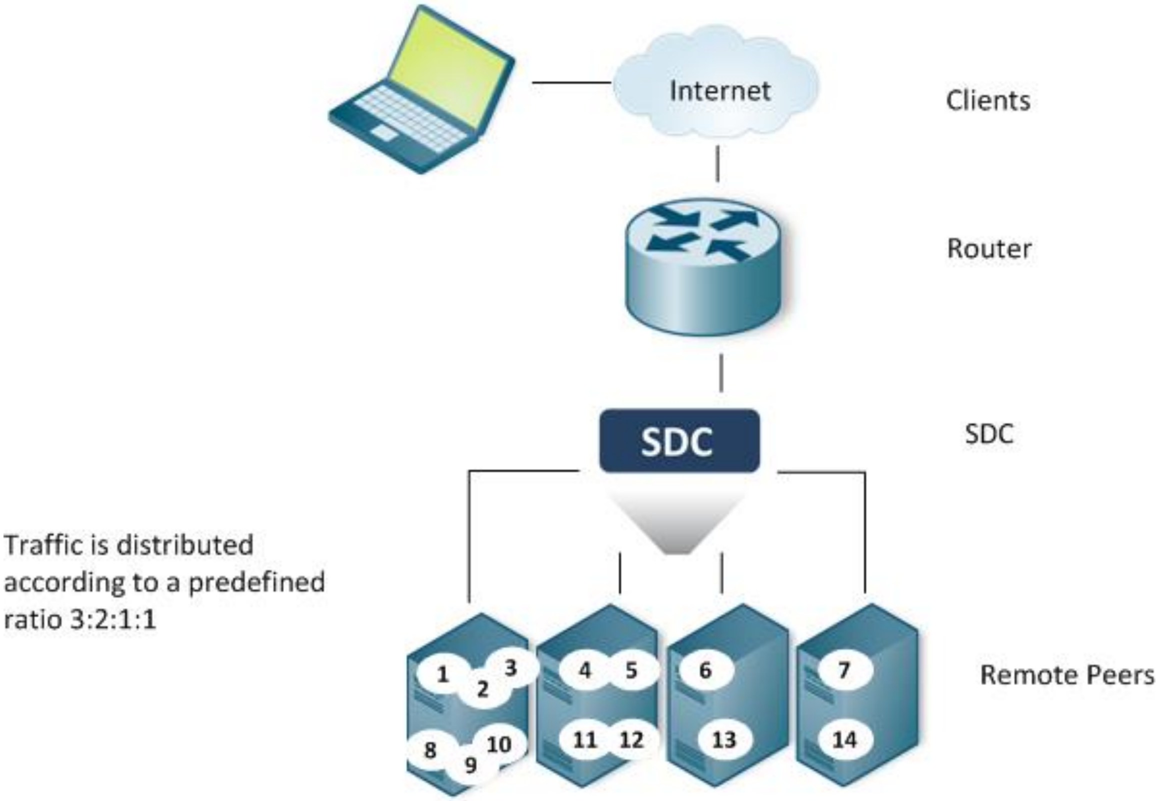

Internet
Clients
Router
SDC
SDC
Traffic is distributed evenly among all Remote Peers in the Pool:
1
5
2
6
3
7
4
8
Remote Peers

3.2.7.3.3 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 23*.

Figure 23: Weighted Round Robin Policy


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

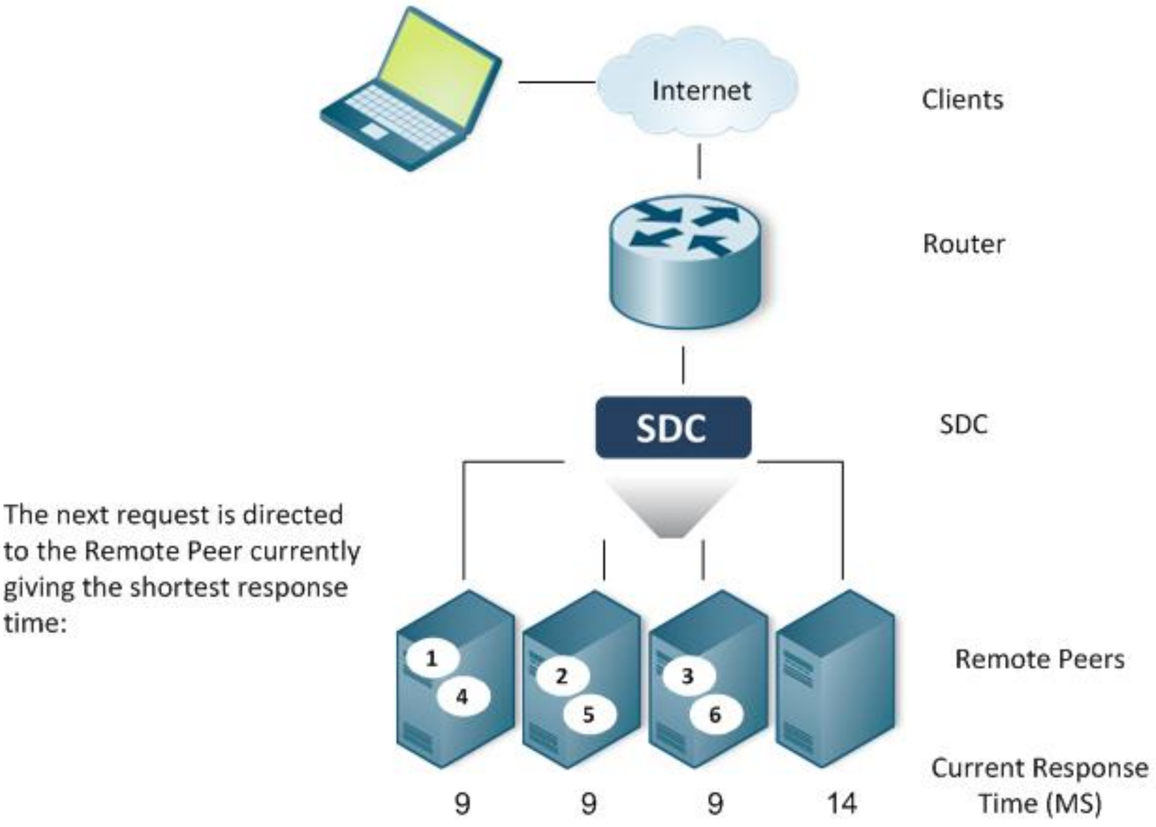
3.2.7.3.4 Fastest Response Time

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 24*.

Figure 24: 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)

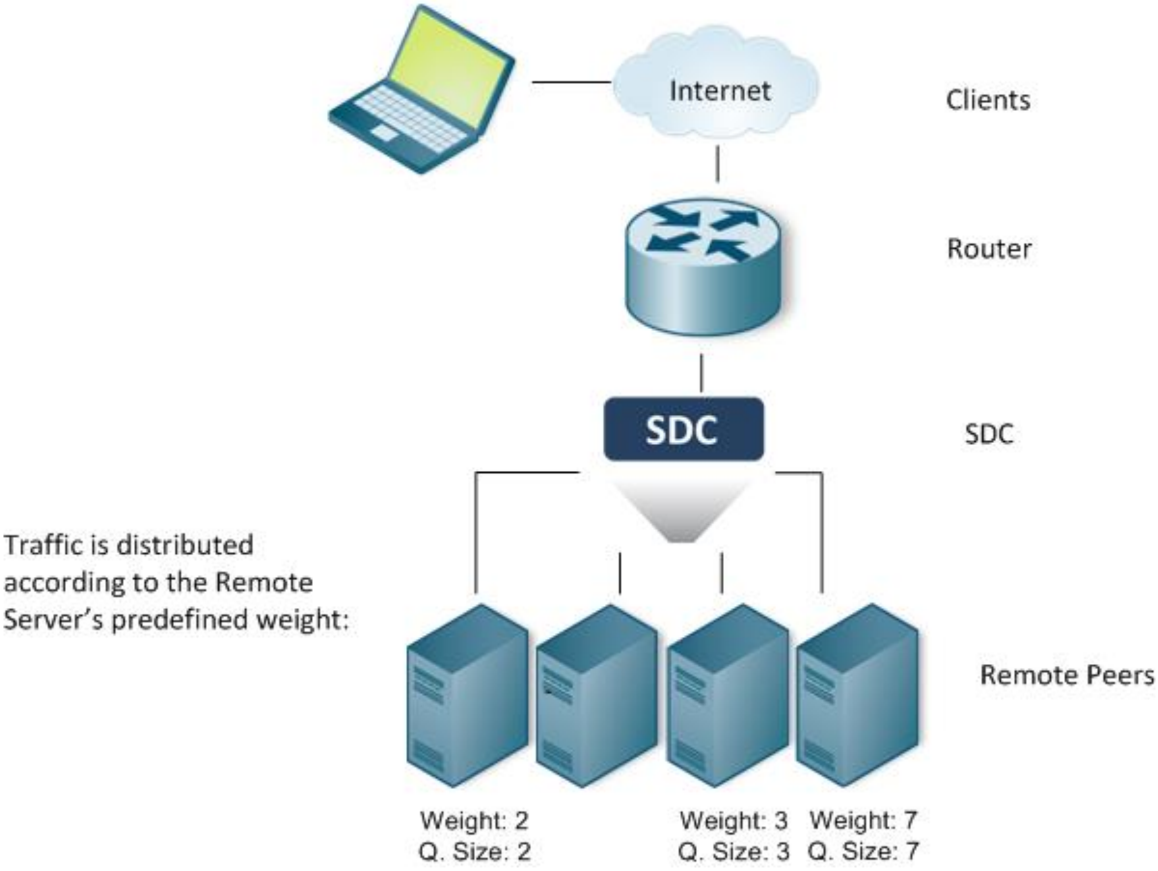
3.2.7.3.5 Queue Size Ratio

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.

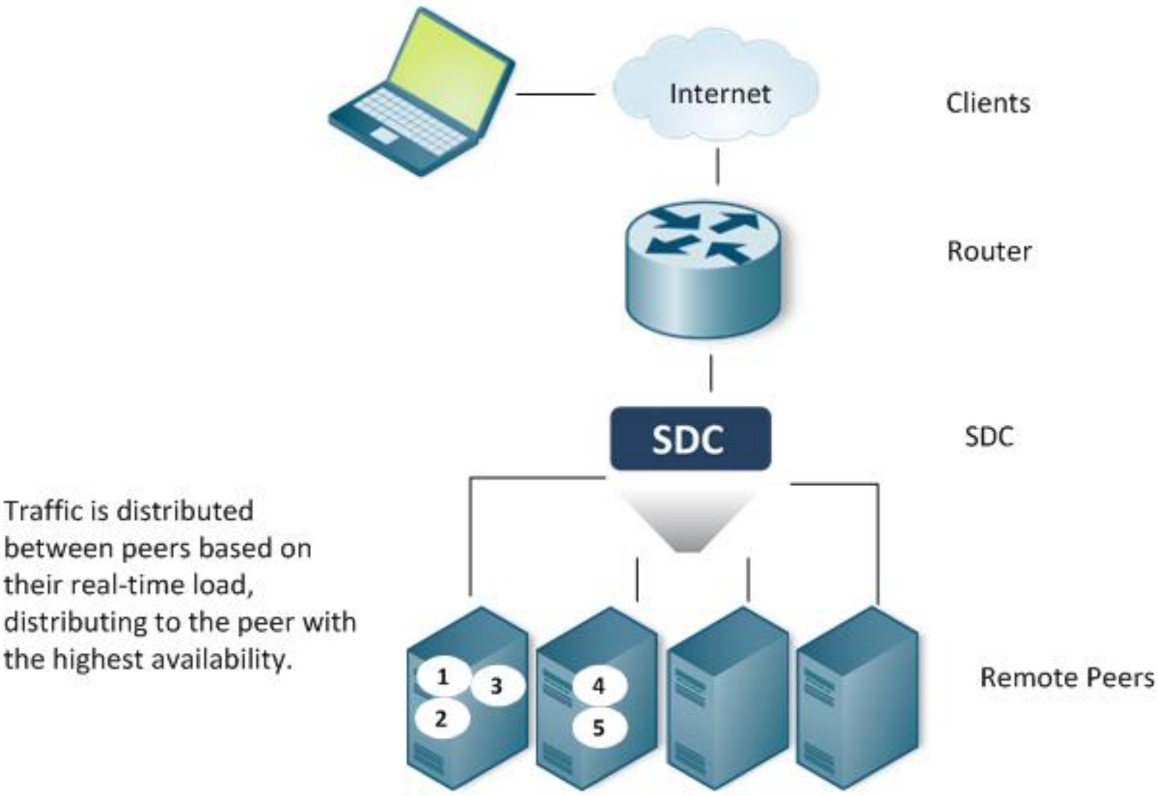
Figure 25: Queue Size Ratio Policy



3.2.7.3.6 Load Based

When selecting a **Load Based** policy, the requests are distributed between servers based on the real-time performance and load experienced by the servers in the pool. Servers with the least load will be the first to receive requests.

Figure 26: Load Based 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

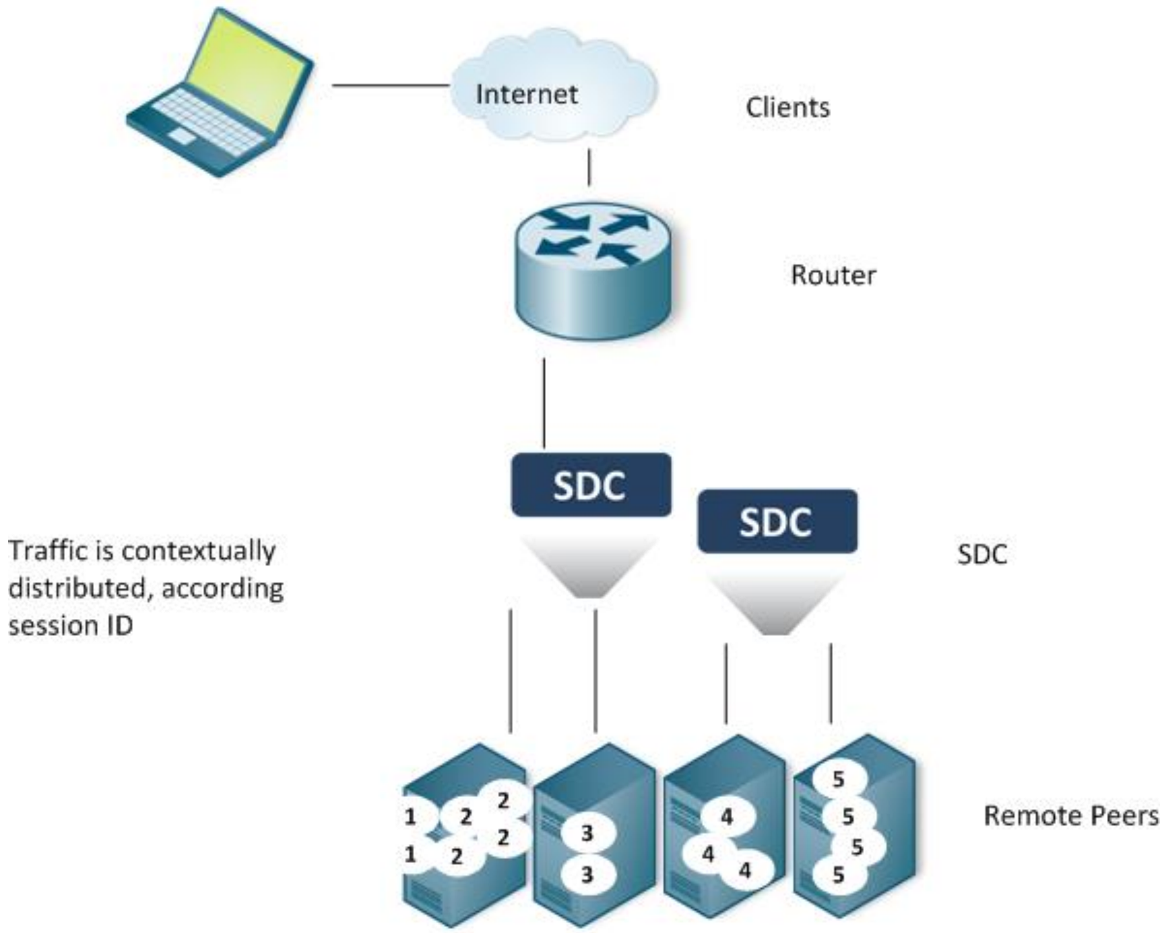
3.2.7.3.7 Contextual

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 27*.

Figure 27: Contextual Policy

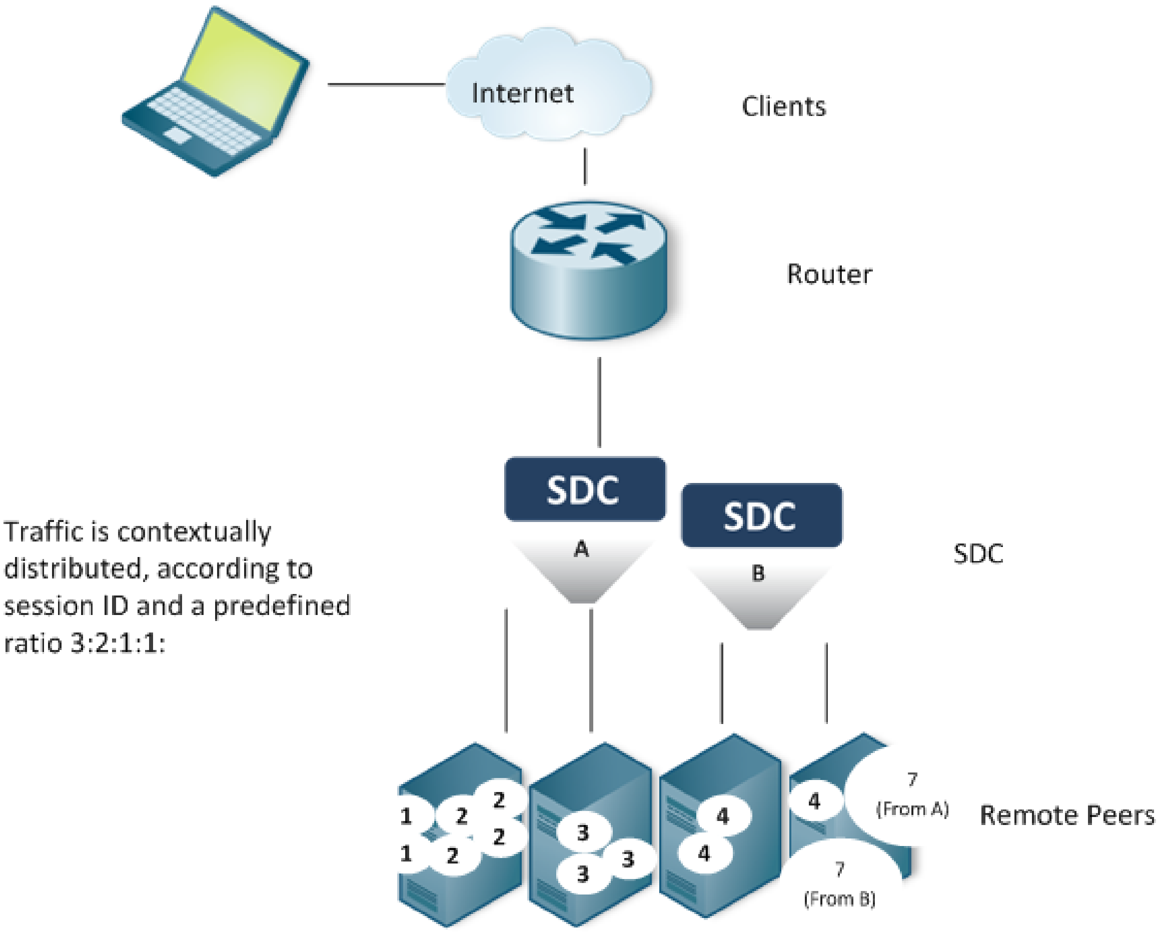


3.2.7.3.8 Weighted Contextual

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 28*.

Figure 28: Weighted Contextual Policy

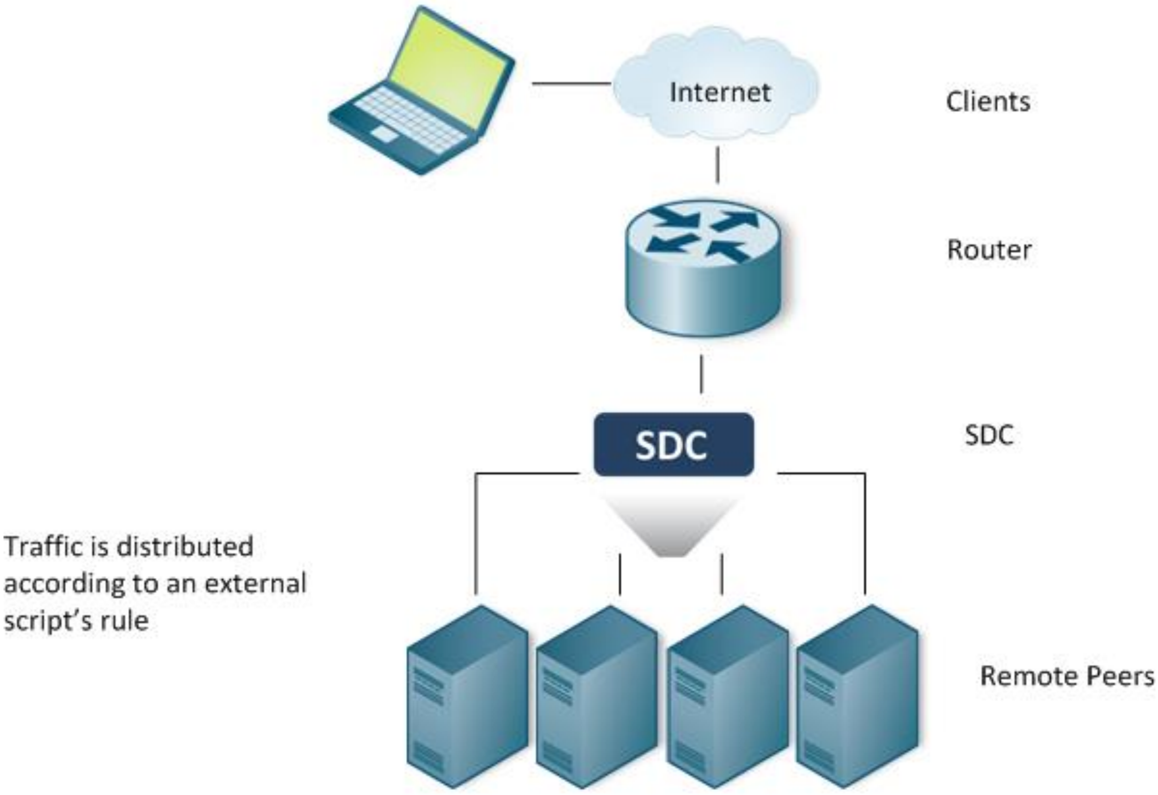

Internet
Clients
Router
SDC
A
SDC
B
SDC
Traffic is contextually distributed, according to session ID and a predefined ratio 3:2:1:1:
1
2
2
1
2
2
3
3
3
4
4
4
7
(From A)
7
(From B)
Remote Peers

3.2.7.3.9 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 29*.

Figure 29: External Policy


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

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

1. From the **Policy** drop-down list, select **External**.
2. 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).
3. In **Peer Selection**, type in the script according to which traffic is distributed across the available Remote Peers.

Table 18 details the parameters that SDC provides to the script:

Table 18: External Script Parameters

External Script's Returned Value Type: Peer	
Parameter	**Type**
Request	Message
peerSelector	PeerSelector
peerTable	Peer Table
acivePeerList	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>
```

3.2.7.4 Editing a Pool

This section describes how to edit a pool.

To edit a Pool:

1. Select a Pool from the list and click **Edit**. The Edit Pool dialog box is displayed:
2. You can edit the enabled fields, as detailed in *Adding a New Pool*.

3.2.7.5 Removing a Pool

You can remove any pool from a site.

To remove a pool:

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

2. Select the Pool from the table.
3. Click **Remove**.

   A confirmation message appears.

4. Click **Yes**.

To refresh the Pool list in the table:

1. Click **Refresh**.

3.2.8 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:

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

To add a rule:

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

2. Under **Address**, enter the IP address of the Client Peer. A CIDR formatted address 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 match the selected Peer Profile.

4. Under **Action**, select whether to **Accept** or **Reject** the Client Peer.
5. Under **Enabled**, select whether this rule is enabled (**True**) or disabled (**False**).

To change the order of the rules:

1. Select the rule from the list
2. 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:

1. Select the rule from the list.
2. Click **Remove**.

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

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

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**.

3.2.9 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 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

3.2.10 **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.

3.2.10.1 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.

3.2.10.2 Response Based Monitor

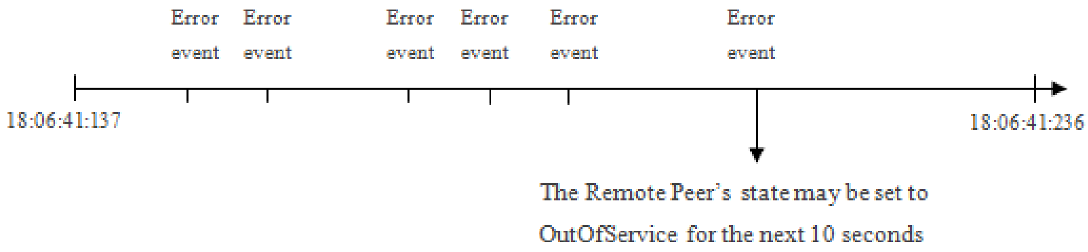
When SDC sends a request to a Server Remote Peer and receives a response that is considered an error, it adds the spotted error to the accumulated number of errors received from that Server Remote Peer.

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 30*.

Figure 30: Error Events in a Measuring Interval



To decide whether to set a peer's state to Out of Service or not, use the following functions in order to receive a peer's statistics within the predefined interval:

Table 19: Statistic Data Functions

Function	Comments
peer.getPendingRequestsCount()	Approximation of the server's number of pending requests.

3.2.10.3 Setting an Error Detection Monitor Parameters

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

3.2.10.4 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.

3.2.10.5 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 – 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** > **Specific Site Settings** > **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 20 details the Health Monitoring Condition Script parameters.

Table 20: 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.

   Table 21 details the Health Monitoring Check script parameters.

Table 21: Health Monitor Check Script Parameters

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

Health Monitor Check Script's Returned Value Type: none	
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 Site Replication

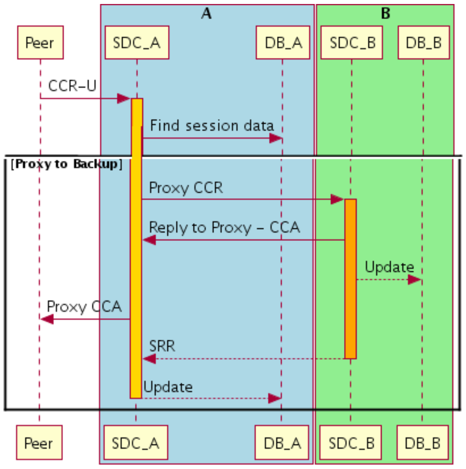
Site replication allows geographically distributed SDC clusters to synchronize session data amongst sites. Session data includes the following:

- Session ID
- Destination Peer
- Pool name
- Origin Peer
- Session Binding data

Session data is distributed by one SDC node (the origin node) to Remote Servers (the target nodes) configured to receive and handle the replicated data.

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, as shown in *Figure 31*.

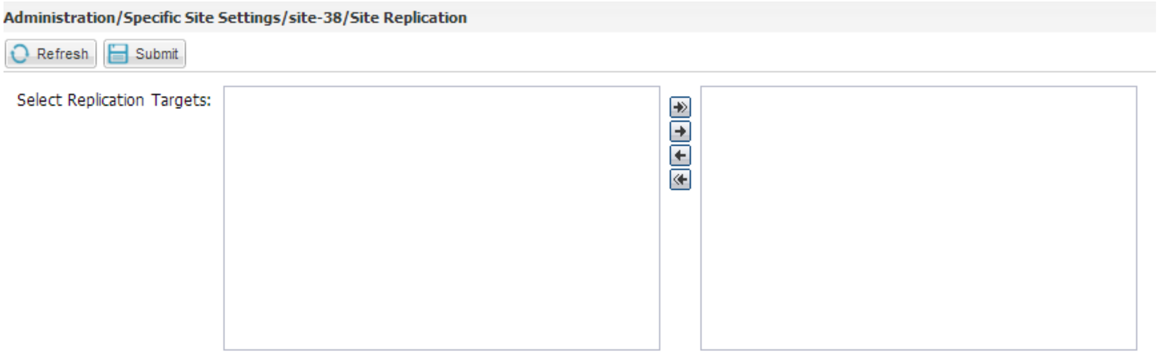
Figure 31: Site Replication



To select the replication targets:

1. Go to **Administration** > **Specific Site Settings** > **Site Replication**.
2. The list on the left contains all the Remote Peers you may select as a target site to handle the replicated data.

Figure 32: Replication Targets


Administration/Specific Site Settings/site-38/Site Replication
Refresh
Submit
Select Replication Targets:

Note: If the list is empty, none of the Remote Peers have been configured to be used for site replication. Setting a Remote Peer to handle replicated data is configured when creating the Remote Peer. For additional information on Remote Peers, see *Adding a New Remote Peer*.

3. Click the **single right arrow** button. The Remote Peer is added to the Replication Target list.
4. Repeat the above steps for each Remote Peer you want to add to the list.
5. To add all available Remote Peers to the target list, click the **double right arrow** button.
6. To remove a Remote Peer from the list, click to select it from the right box and then click the **single left arrow** button. To remove all Remote Peers from the list click the **double left arrow** button.

Note: The selected Remote Peers will handle replicated session data, only if you select the **Use for Geo Redundant Sites Connection** checkbox in the **Remote Peers** configuration (**Topology** > **Site Settings** > **Site** > **Remote Peers** > **Add Server Peer**) and if you save the relevant session data for replication (**Routing** > **Session Management** > **Session Properties** > **Session Persistence Policy** > **Persist and Replicate**).

7. Click **Submit**.

3.4 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. The Rate Limits control the data flow and prevent data from overloading the system by setting byte and message send and receive thresholds. Each Remote Peer must operate below these thresholds in order to be actively connected to SDC. In addition to the channel level threshold, global rate limits may also be defined.

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.

To change the Default Transport Configuration:

1. Go to **Administration** > **Specific Site Settings** > **Default Transport Configuration**. The Default Transport Configuration screen displays the **Socket Defaults** tab.
2. From the **SDC component** drop-down list, select the **CPF/FEP** Node that you want to apply the Default Transport Configuration changes.

Table 22 details the Socket Defaults and their descriptions:

Table 22: Socket Defaults

Parameter	Description
Buffers (TCP and SCTP)	
Send Buffer Size	The TCP and SCTP sending buffer size (for outgoing data).
Receive Buffer Size	The TCP and SCTP receiving buffer size used (for incoming data).
Socket Options (TCP and SCTP)	
TCP No Delay	Disable Nagle's algorithm for this connection. Written data to the network is not buffering pending acknowledgement of previously written data.
So Linger	Specifies the timeout for brute-force shutdown of a channel, after a close request (TCP level) is sent from SDC to a remote node.
TCP	

Parameter	Description
Keep Alive	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.
Reuse Address	When enabled, used for MulticastSockets in java, and it is set by default to True for MulticastSockets.
Traffic Class	This option sets the type-of-service or traffic class field in the IP header for a TCP or UDP socket.
SCTP	
Heartbeat Interval	This is the interval when a HEARTBEAT chunk is sent to a destination transport address to monitor the reachability of an idle destination transport address.
Cookie	Handle COOKIE PRESERVATIVE parameter in the INIT chunk.
Number of Inbound Streams	The number of SCTP inbound streams.
Number of Outbound Streams	The number of SCTP outbound streams.

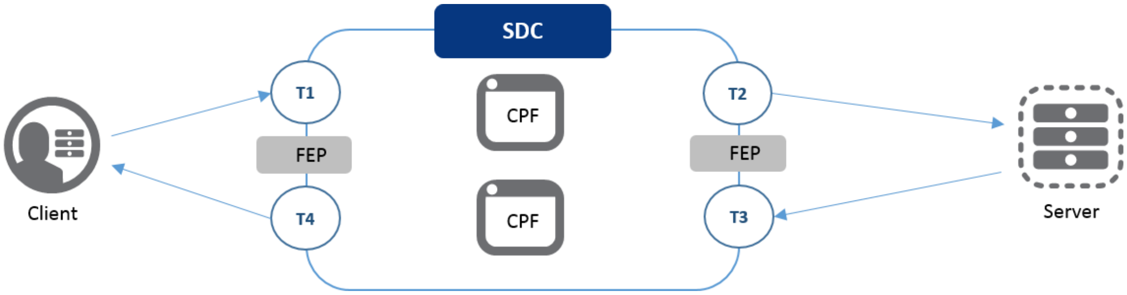
Parameter	Description
Support Unordered Delivery	Enable support for accepting and processing SCTP data chunks as they arrive, even if they are out of sequence.
SCTP Profiles	
SCTP Profile	The SDC contains the following preconfigured SCTP profiles. Each profile is configured with pre-defined parameters. Select one of the following profiles or select **Custom** to configure a unique SCTP profile: Default Same US State US Coast to Coast and Inside EU Asia-Asia EU-USA EU/US - Asia Universal
Association Max Retrans	Maximum number of retransmission attempts to a peer per association, by message type.
Pa**th Max Retransmits**	Maximum number of retransmission attempts to a peer per path, by message type.
RTO Initial	The initial value of RTO (retransmission timeout) that is used in RTO calculations.
Max Init Retransmits	Maximum number of attempts to establish a path connecting to a peer.
RTO Min	Minimum value used for the RTO. If the computed value of RTO is less than RTO Min, the computed value is rounded up to this value.

Parameter	Description
SCTP_MAXSEG	Maximum size of the data chunks that the SCTP message can be divided into for all the paths in an association.
RTO Max	Maximum value 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	Time in milliseconds that the peer waits for a selective acknowledgement (SACK).

3.4.1 Configuring Rate Limits

Rate limits are configured to control the amount of traffic that the SDC node receives from either a client or server peer and/or sends towards servers or a pool of servers. These limits are configured by the number of messages and/or bytes that the SDC can receive and/or send.

Figure 33: Basic Traffic Flow between the SDC and Networks



The basic traffic flow between the SDC and the networks is illustrated in *Figure 33*. 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 and can be limited. These limits ensure that the overall traffic flow performance is constantly under control and no service degradation will occur in overload conditions.

3.4.1.1 Configuring the Incoming Traffic Rate Limits

The incoming traffic rate limits are configured to control the amount of traffic that the SDC node 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. This incoming traffic can either be limited per the client or server peer that the traffic is sent from, or per the SDC component (FEP/CPF) that receives the traffic.

To configure the Rate Limit per SDC Component:

1. Go to **Administration** > **Specific Site Settings** > **Default Transport Configuration** > **Rate Limit**.
2. Select the SDC component from the drop-down list.

Note: While each FEP can be configured with a unique rate limit, all CPFs are configured with the same rate limit.

3. Fill in the values for the rate limits and select the desired overload policy, as detailed in *Table 23*.

Table 23: Rate Limit

Rate Limit/Overload Policy	**Description**
Global Rate Limits	
Byte receiving rate limit (from all peers)	Sets the limit of bytes that can be received per second from all client/server peers
Message receiving rate limit (from all peers)	Sets the limit of messages that can be received per second from all client/server peers
Read Limit Policy	Defines the behavior of the SDC once the rate limits have been exceeded. The selected policy is invoked when either the global or peer rate limit has been exceeded.

Rate Limit/Overload Policy	Description
	▪ **Discard With Answer** – discards all incoming messages and returns a busy Result-Code (default) or other configurable result codes (MessageDefaultErrorCode) ▪ **Silent Discard** – discards all incoming messages and does not return any answer
Peer Rate Limits	
Note: Incoming rate limits should be configured under the relevant SDC component (CPF or FEP) that the peer is connected to. Outgoing rate limits should be configured under the relevant CPF.	
Byte receiving rate limit (per peer)	Sets the limit of bytes that can be received per second per peer
Message receiving rate limit (per peer)	Sets the limit of messages that can be received per second per peer
Overload Policy	▪ **No Overload Policy** – sets if to apply overload policy for a peer ▪ **Message sending rate limit (per peer)** - sets the limit of messages that can be sent per second per peer (server/client)

To configure the Incoming Traffic Rate Limit per Peer:

1. In **Step 2: Peer Profile Configuration** in the **Add Peer Profile** wizard, click **Rate Limits**.
2. In the **Peer receive rate limits** area, fill in the values for the rate limits and select the desired overload policy, as shown in *Table 23*.

3.4.1.2 Configuring the Outgoing Traffic Rate Limits

The outgoing traffic rate limits are configured to ensure that the peers and server pools that the SDC sends messages to can efficiently receive the messages. When a peer (or pool of server peers) nears or exceeds the configured rate limits, the traffic sent to it by the SDC is prioritized and minimized, to ensure that minimal server degradation is experienced.

To configure the Outgoing Traffic Rate Limit per Peer:

1. Go to **Step 2: Peer Profile Configuration** > **Add Peer Profile** > **Rate Limits**.
2. In the **Peer send rate limits** area, select the **Message sending rate limit (per peer)** checkbox and enter the desired rate limit in the field below.
3. Click **OK**.

To configure the Outgoing Traffic Rate Limit per Pool:

1. Go to **Add Pool** > **General**.
2. In the **Rate Limit (TPS)** field, enter the desired rate limit.
3. Click **OK**.

3.4.2 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 a Tripo repository once the session destination is determined. All session data is replicated and synchronized in Tripo instances either within an SDC site or in Tripo 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.

Note: To replicate between Tripo instances on mated SDC sites, the Tripo Site Replication feature must be enabled during the installation or upgrade process, as described in the *F5 SDC 4.4 Installation/Upgrade Guide*.

3.5 Licensing the VIPs

Prior to the FEPs being able to process traffic, each VIP address that is associated with a configured FEP must be given a separate license key. The license key is generated and provided to you by F5 Support.

3.5.1 Adding a New License Key

Each new license key needs to be added.

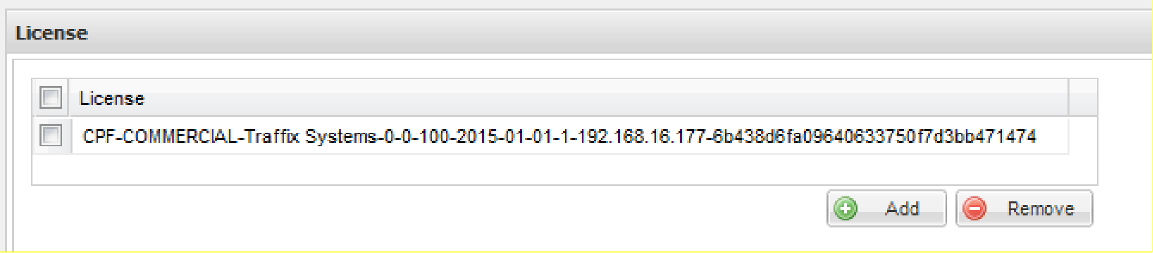
To enter a new license key:

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**.

3.5.1.1 The License Key's Structure

The provided key represents different properties, separated by a hyphen, as shown in *Figure 34*.

Figure 34: License Key



License

□	License
□	CPF-COMMERCIAL-Traffix Systems-0-0-100-2015-01-01-1-192.168.16.177-6b438d6fa09640633750f7d3bb471474

Table 24 describes the license key properties:

Table 24: 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
192.168.16.177	IP Address

Note: Multiple license keys can support multiple IP addresses used by SDC.

3.5.2 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 **Yes**.

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 35*and detailed in *Table 25*.

Figure 35: SDC Internal Flow Logic

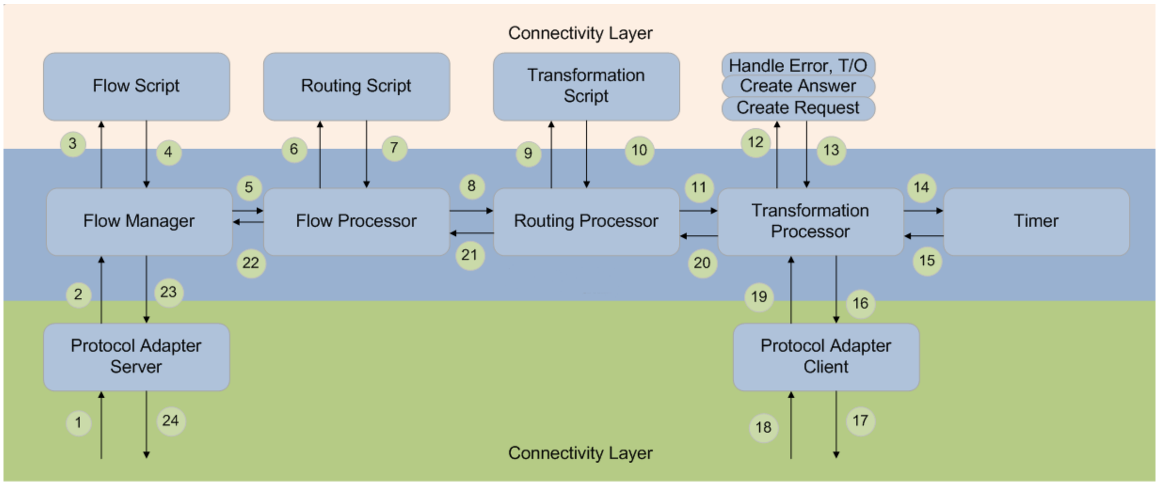


Table 25: SDC Flow Logic Legend

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.

To set the script actions for the routing and transformation actions, one must 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).

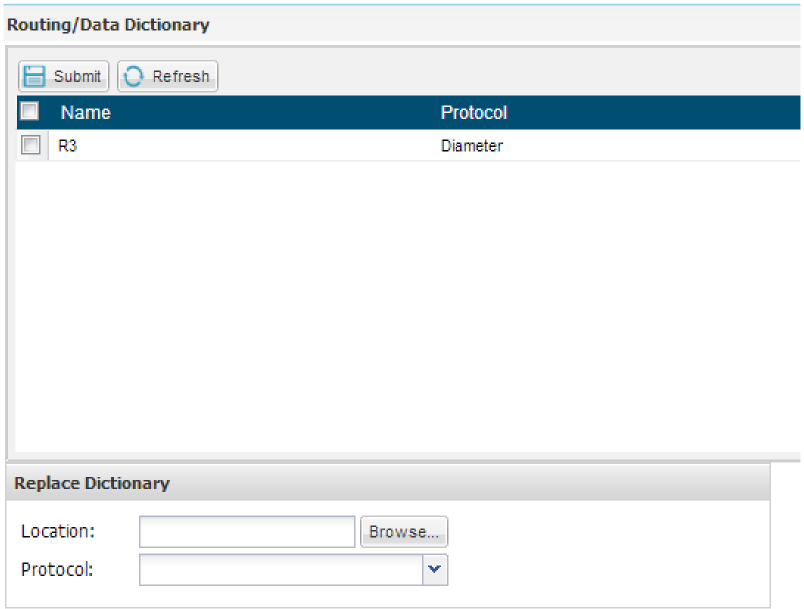
4.1 Dictionary

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 replace the selected data dictionary:

1. Go to **Routing** > **Data Dictionary**. The Data Dictionary screen displays the currently selected data dictionary.

Figure 36: Data Dictionary



☐	Name	Protocol
☐	R3	Diameter

2. Click **Browse** and select the data dictionary file's location.

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

4. Click **Submit**. The newly selected data dictionary appears in data dictionary list.

4.2 External Lookup Management

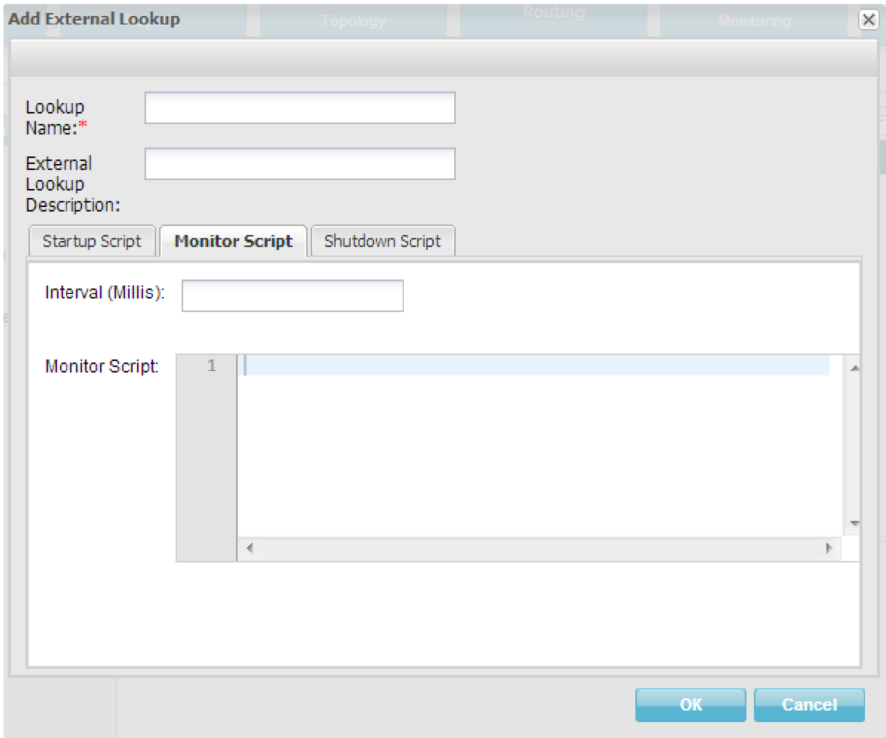
External lookup items allow you to run scripts to extract data from external sources such as LDAP or Coherence. You may define scripts to run upon SDC startup and shutdown which will obtain information that can be used by SDC. You may use external lookup scripts in Session Binding, for example.

To add an external lookup item:

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

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

Figure 37: Add External Lookup



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

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

5. 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;
```

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

```
//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 26 details the External LookupScript parameters.

Table 26: Lookup Script Parameters

Parameter	Type
Stack	Stack
externalLookupProperties	PropertiesOwner
UserTraceLoggerWrapper	userTraceLogger
metaData	MetaData

Note: You may only call API methods associated with the parameters listed in *Table 26* (to view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

8. Click **OK**. The new External Lookup item is added. You may click the item’s line to edit it.

4.2.1 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 **Routing** > **External Lookup Management**, select a row in the table.
2. Click **Disable**.

   A confirmation message appears.

3. Click **Yes**.

4.2.2 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:

4. Select the row of the External Lookup data source that you want to remove.
5. Click **Remove**.

   A confirmation message appears.

6. Click **Yes**.

4.3 Session Management

As an SDC Web UI administrator, you can apply a session binding 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 a rule-based session binding. The binding rules consist of parameters that are defined by the Session Binding Attributes – a list of AVP’s. Each AVP is assigned a type (Boolean, regular expression, etc.).

The session binding 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.

The session binding is done using Binding Keys. Binding 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.

The Session Binding Attribute list may, for example, consist of the following AVP's: Request Type, and IMSI (International Mobile Subscriber Identity). When setting the Session Binding rules you may use these AVP's to determine whether the session match a specific Request Type, and a specific IMSI. When a matching session is found, the selected session binding method is applied.

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.

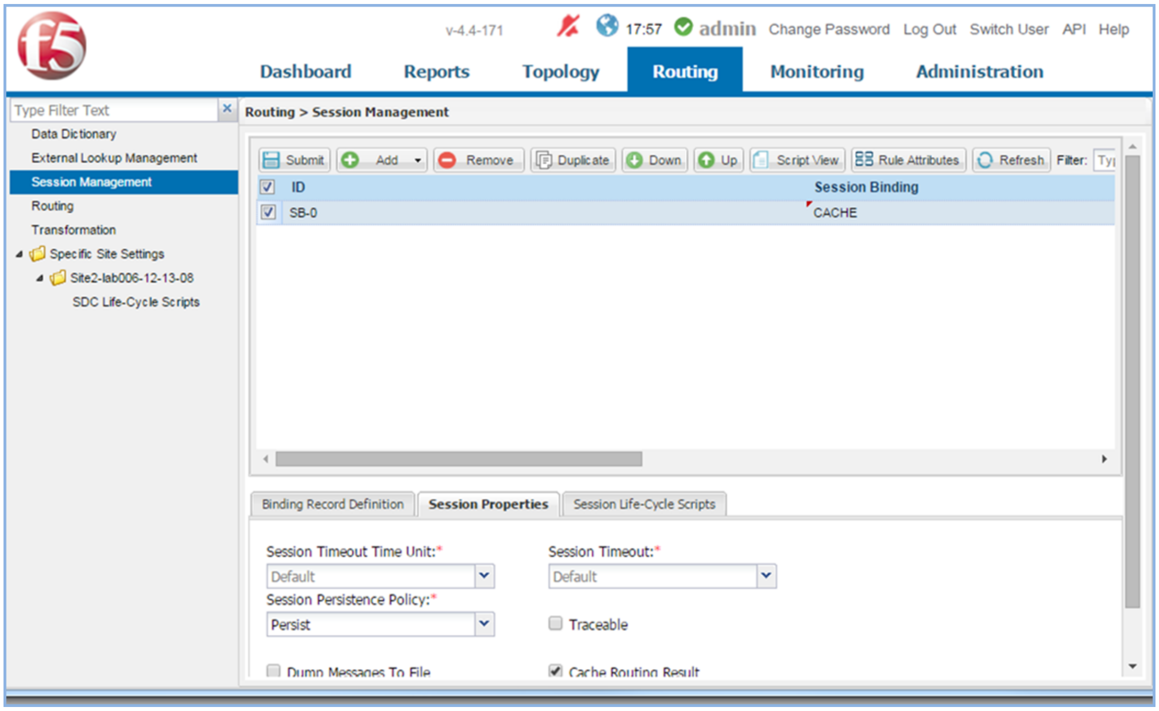
4.3.1 Configuring Session Binding Rules

The following section describes how to configure Session Binding rules and attributes.

To configure a session binding rule:

1. Go to **Routing** > **Session Management**. The Session Management screen is displayed.

Figure 38: Session Management



☑ ID	Session Binding
☑ SB-0	CACHE

Session Timeout Time Unit:* Default

Session Timeout:* Default

Session Persistence Policy:* Persist

☐ Traceable

☐ Dump Messages To File

☑ Cache Routing Result

4.3.2 Adding a Session Binding Attribute

Prior to configuring a rule based session binding rule, you must configure parameters that are defined as in Session Binding Rule Attribute.

To add a Session Binding attribute:

1. Click **Rule Attributes**. The Session Binding Attribute window appears.
2. Click **Add**. A new line representing an attribute is added to the table.
3. Under **Label**, type in a user friendly name that will be used to identify the attribute. For example: “Application ID”.
4. Under **Attribute**, type in the name of the AVP retrieved from the message.
5. Under **Type**, select the data type of the new attribute.
6. Under **Description**, type in a short description of the attribute.
7. Click **Submit**.

Note: For additional information on the decision table attributes, see *Appendix D: Decision Table Attributes*.

4.3.3 Adding a Session Binding Rule

This section describes how to add a Session Binding rule.

To add a Session Binding rules:

1. Return to the **Session Management** screen. The table's columns represent the Session Binding Attributes you previously defined. If you have not yet set any attributes, see *Configuring Session Binding Rules*.
2. Click **Add**. A new row with an automatically assigned rule name (**ID**) is added to the table.

Note: If you remove a Session Binding rule from the table, the next added rule is assigned an ID name based on continued numbering of the previously removed rule.

3. Under each column reflecting the added attributes, select the value against which new sessions are compared. For example, under **OHost** set the value client_Rf-1.
4. Under **Session Binding**, select the session binding method to invoke upon a matching session identification.

Table 27 describes the different session binding methods and their configurations:

Table 27: Session Binding Methods

Session Binding Rule	Description	Method Configuration
Cache	Cache is the default option and indicates that the Routing is performed based on the routing rule and the routing decision creates a binding record entry holding the relevant keys.	Binding Record Selection Session Properties Session Life Cycle Scripts

Session Binding Rule	Description	Method Configuration
	You must specify the key sets (zero or more) that can be used for resolving this binding record.	
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 script and the key sets (zero or more) that can be used for resolving this binding record.	External Lookup Script Binding Record Selection Session Properties Session Life Cycle Scripts
Resolve	Indicates that cached routing decisions should be used for this session. Routing will not be performed if cached routing decision do not exist. 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 will always check Tripo for its master's state (based on its binding key) and never for the state of the session (based on its session ID).	Binding Key Selection Session Properties Session Life Cycle Scripts
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 Script Binding Key Selection Session Properties Session Life Cycle Scripts

Session Binding Rule	Description	Method Configuration
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
No Binding	Indicates no binding. In this case only the Life-Cycle scripts are applied to the matching session.	Session Properties Session Life Cycle Scripts

4.3.3.1 Defining Binding Keys

A binding record includes the binding keys related to a selected session. A binding key consists of a name and value. 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 binding keys are saved in the Tripo repository. You only define the binding keys when you select a **Cache** (Master) or **External** Binding Rule. The session ID is always the first binding key and you can add up to four other keys, such as IPV6 or an IMSI.

To add a binding 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**.

4.3.3.2 Selecting a Defined Binding Key

When configuring a **Resolve** binding rule, you need to select one of the binding keys that was defined for a related Cache or External session rule.

To select a defined binding key:

1. Click **Binding Key Selection**.

2. In **Defined Keys**, select from the drop-down list the key against which you want the resolved session to be compared.

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.

4.3.3.3 Defining Session Properties

This section describes how to configure the session properties.

To configure the session properties:

1. In **Session Timeout Time Unit (Seconds/Minutes/Hours/Days/Weeks)**, from the drop-down list, select the time resolution of the timeout parameter.

    a. In **Session Timeout**, from the drop-down list, select the 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.

2. In **Session Persistence Policy**, select one of the following options:

- **Persist** - to save the session data in a Tripo repository in a single site SDC deployment
- **Non Persistence** - to not save the session data
- **Persist and Replicate** - to save the session data in a Tripo repository and replicate it to another Tripo instance on an SDC mated site

Note: If you select **Non Persistence**, 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 Tripo Site Replication feature, you must select **Persist and Replicate**.

The Tripo Site Replication feature must be enabled during the installation or upgrade process. For more information about saving and replicating session data, see *Replicating Session Data*.

Do not **Persist** HTTP sessions.

3. Select **Dump Messages to File** to trace the session, collect its data and display it in Transaction Data Records, in Reports.
4. Select **Release Upon Session Termination Event** to release Diameter session 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).
5. Select **Traceable** to trace the session using log messages. Log messages are printed to the log, per traceable log, upon message sending or retrieval and before script invocation in Debug level.
6. Leave the **Cache Routing Result** checkbox selected. This causes the routing decision for this session to only be made once per session – during session initialization. This routing decision will be applied for all subsequent events of this session. Clearing the **Cache Routing Result** checkbox will cause a new routing decision to be made for each event in the session, as a destination peer is selected from a pool.

Note: Selecting and clearing the **Cache Routing Result** checkbox does not have any effect on the selected session persistency policy.

7. Select **Idle Session Timeout** to reset a timeout upon every session data withdrawal in addition to resetting the session timeout regularly upon session data update.

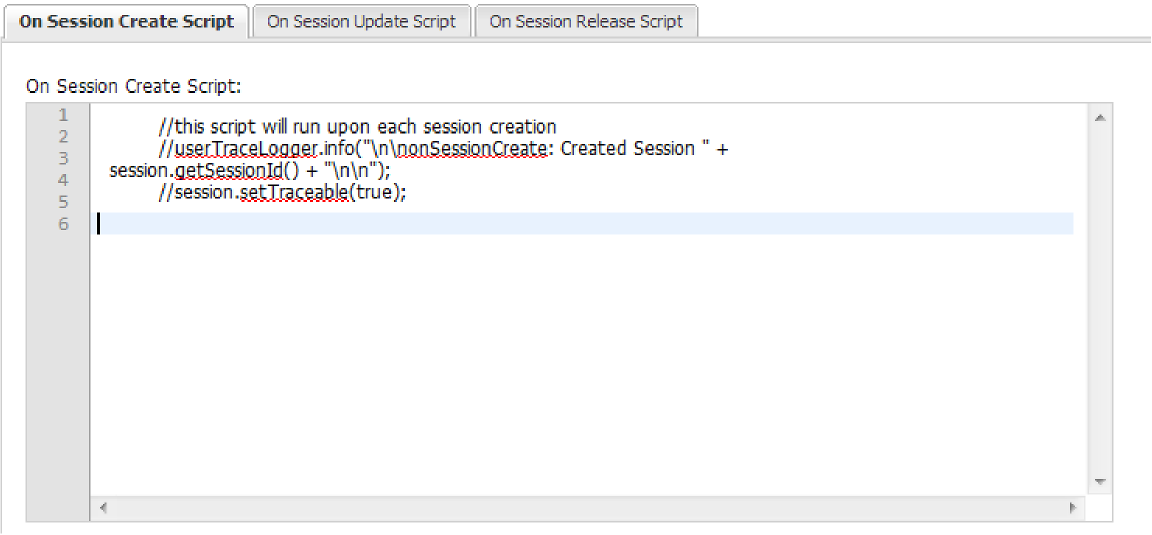
4.3.3.4 Configuring Session Life-Cycle Scripts

Life-Cycle scripts run upon session creation, session update, and session release. 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 39: On Session Create Script



```
        //this script will run upon each session creation
        //userTraceLogger.info("\n\nonSessionCreate: Created Session " +
session.getSessionId() + "\n\n");
        //session.setTraceable(true);
```

2. Set the script to run each time a new session is created.

Table 28 details the parameters SDC provides to the scripts.

Table 28: 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:

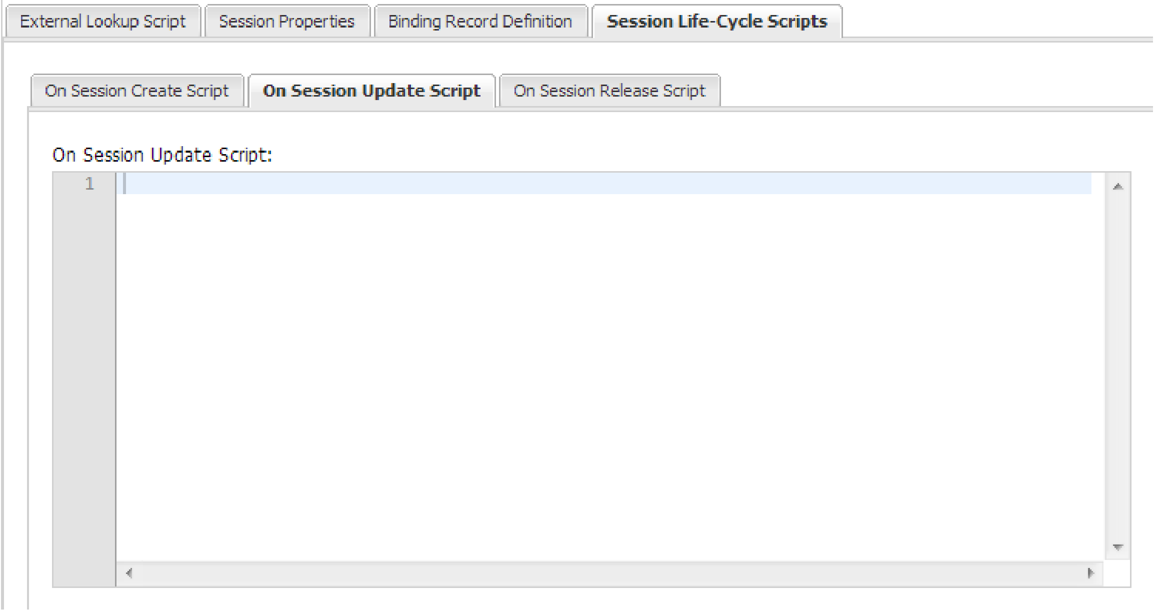
```
def vasId = message.get("ServiceInformation").get("MMSInformation").get("VAS-
ID").get();

if (vasId.equals("MMS")) {
session.setTraceable(true);
}
```

To implement the On Session Update script:

1. Click **On Session Update Script**.

Figure 40: On Session Update Script


External Lookup Script
Session Properties
Binding Record Definition
Session Life-Cycle Scripts
On Session Create Script
On Session Update Script
On Session Release Script
On Session Update Script:

2. Set the script to run each time a new session is created.

Table 29 details the parameters used in the On Session Update script.

Table 29: 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 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 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);
       }
        }
```

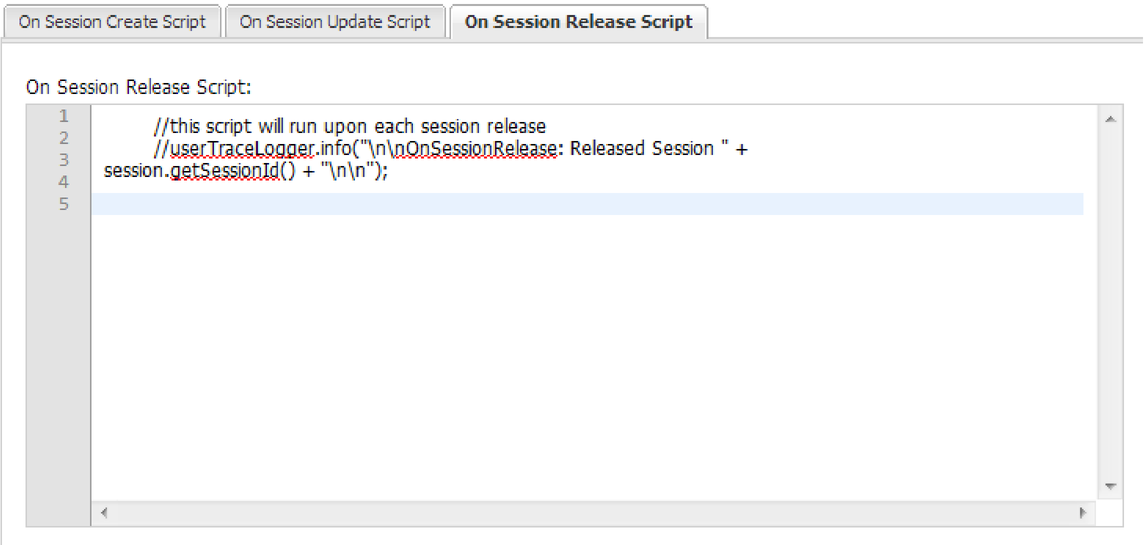
```
      } else
     userTraceLogger.log(Level.WARN, " No action -> Destination peer wasn't
changed");
    } else {
  userTraceLogger.log(Level.WARN, " no key was found for the session:" +sessionId);
  }
  return null;
   ]]>
  </OnSessionUpdate>
```

To implement the On Session Release script:

Note: This script is called only upon calling session.release() method, and not upon session timeout.

1. Click **On Session Release Script**.

Figure 41: On Session Release Script



On Session Release Script:

```
        //this script will run upon each session release
        //userTraceLogger.info("\n\nOnSessionRelease: Released Session " +
session.getSessionId() + "\n\n");
```

2. Set the script to run each time a session is released.

Table 30 details the On Session Release Script Parameters.

Table 30: 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 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 Release Script:

```
def vasId = message.get("ServiceInformation").get("MMSInformation").get("VAS-
ID").get();

if (vasId.equals("MMS")) {
userTraceLogger.trace("Done with session " + session.getSessionId());
}:
```

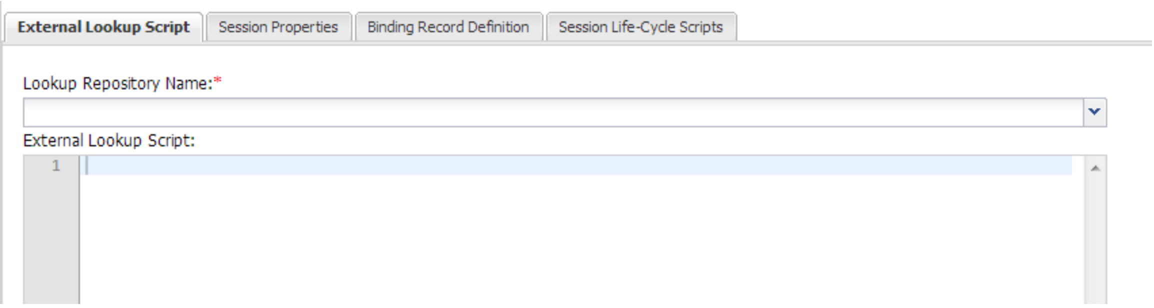
4.3.3.5 Configuring the External Lookup Script

The External Lookup Script is the external script that is applied when **External**/**External or Cache** binding method is selected. The script defines the way to handle the session.

To define the External Lookup Script:

1. Click **External Lookup Script**.

Figure 42: External Lookup Script



2. In **Lookup Repository Name**, select an external repository from the drop-down list.

   For information on external lookup scripts, see *External Lookup Management*.

3. Type in the script.

Table 31 details the parameters SDC provides to the scripts:

Table 31: 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 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 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.");
                }
```

4.4 Routing Mechanism

The SDC Routing mechanism 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.

Routing objects are sets of rules according to which SDC routes messages to the correct destination. Each routing rule within the Routing table contains a set of parameters, corresponding with the message’s content. That is, SDC determines how to treat each message, based on its content. Each parameter is represented by a Rule Attribute – an AVP. The Rule attributes are configured independently and each AVP is assigned a type (Boolean, regular expression, etc.)

The Rule Attributes list may, for example, consist of the following AVP's: applicationId, isRequest and DestinationHost. When setting the actual routing rules you may use these AVP's to determine the action to be taken when a message's applicationId is 500, and the message is a request and its destination host is empty, or the action to be taken when the message is not a request and the applicationId is 400. The combination of the routing rule's conditions is of type And.

Routing rules are scanned in the order they are listed. The first routing rule's condition that is met (that is, the message's attributes match the rule's criteria), causes the rule's action execution. Based on its content, a message can be routed, forwarded, rejected, discarded, etc. According to the selected action, different scripts are performed.

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 VIPs*.

Configuring the Routing mechanism includes the following procedures:

- *Configuring a Routing Rule*
- *Defining Routing Rule Attributes*
- *Defining Routing Rules*

4.4.1 Configuring a Routing Rule

Routing Rules define how the SDC processes the messages it receives and are configured in a Decision Table. Each routing rule, contains one or more Rule Attributes, which you must also define as part of configuring a routing rule. When a message is received, its property values are checked against the values defined for the Rule Attributes to decide which routing rule to use to route the message.

For more information about configuring a decision table and the associated rule attributes, see *SDC Decision Tables*.

To configure a routing rule:

1. Go to **Routing** > **Routing**. The Routing screen is displayed.
2. Define the relevant rule attributes.

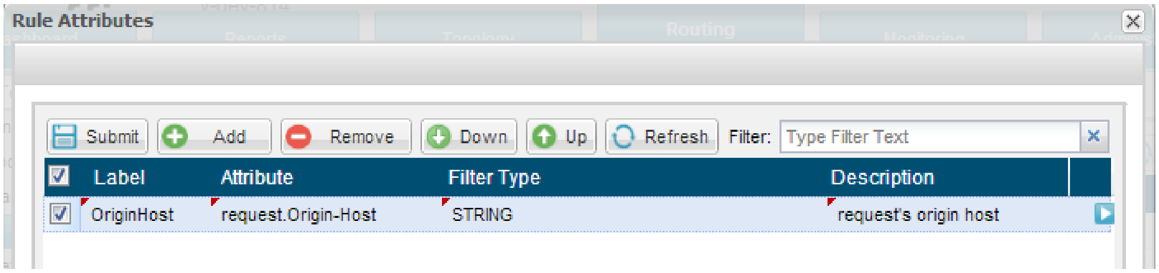
4.4.1.1 Defining Routing Rule Attributes

As part of defining a routing rule, you need to define its rule attributes.

To add a routing rule attribute:

1. Click **Rule Attributes**. The window displays the list of attributes (AVP's) that may be used to define the routing rules:

Figure 43: Rule Attributes



☑	Label	Attribute	Filter Type	Description
☑	OriginHost	request.Origin-Host	STRING	request's origin host

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 **Filter Type**, select the data type of the new attribute. e.g.: String, from the drop-down list.
6. Under **Description**, type in a short description of the attribute.
7. Click **Submit**.

Note: For additional information on the decision table attributes, see *Appendix D: Decision Table Attributes*.

4.4.1.2 Defining Routing Rules

Each Routing Rule is associated with an Action that has different configuration parameters and scripts.

Note: Prior to defining a routing rule action, you need to define rule attribute. For more information on defining rule attributes, see *Defining Routing Rule Attributes*.

To add a Routing Rule:

1. Go to **Routing** > **Routing** > **Add**. A new rule line assigned with an automatic name is added to the table.
2. Under each column, representing the previously defined Rule Attributes, define a value against which new messages are compared. For example: under Application ID set the value "500", under Is Request set the value "True" and leave the Destination Host empty. This rule shall apply to requests of the application ID 500, of which the destination host is empty.
3. Under **Action**, select an **Action** from the drop-down list.

Table 32 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 32: Action Descriptions

Action	Description	Available Configuration Parameters/Scripts
Route	Routes the request to one of the specified Pools.	**Rule Configuration** **Diameter Identity** **TDR Configuration**

Action	Description	Available Configuration Parameters/Scripts
		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**
Reject	Performs a local termination with an error result (the result code should be configured).	**Rule Configuration** **Diameter Identity** **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** **Diameter Identity** **Create Message Locally**

Action	Description	Available Configuration Parameters/Scripts
Resolve & Route	Resolves and routes the request by a designated DNS server	**DNS Resolving** **Row Specific Configuration** **Diameter Identity** **Check error in answer** **Handle Server Error** **Handle Client Error** **Handle Locally**

4.4.1.2.1 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.

To configure the Rule Configuration parameters:

1. Under **Routing** > **Routing**, 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 33*.

Table 33: Routing Rule Configuration Parameters

Parameter	Definition	Default Value	Note
Max Resend Attempts (**Forward**, **Route**, **Resolve & Route**)	Set the maximum number of request sending retries, in case it fails	0	This parameter affects the entire Pool.
Forward to (**Forward**)	Set if the attempt is resent to a **Peer** or **Pool**	**Peer**	
Delay Between Attempts (**Forward**, **Route**, **Resolve & Route**)	Set the time difference between one resend attempt and another	0	
Destination Peer Name (**Forward**)	Set the peer to which the request is sent		
Redirect Host Usage (**Redirect**)	Set the answer’s RedirectHostUsage AVP value		

Parameter	Definition	Default Value	Note
Redirect Max Cache Time (Redirect)	Set the answer's RedirectMaxCacheTime AVP value		
Reject Code (Reject)	Enter the result code returned to the message's origin upon rejection		
Pools (Route)	Select the pool/s to which messages which match the rule's criteria are sent		
Peer Profile (Resolve & Route)	Selects the Peer Profile to create a temporary peer and a temporary pool	Default	
DNS Resolving (Resolve &Route)	Configure the parameters so that the request is resolved by a designated DNS Server and then routed to the relevant server		

4.4.1.2.2 Defining Diameter Identity Parameters

When defining the peer profiles for Diameter peers, 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.

In the **Diameter Identity** tab, you define the Diameter Identity policy for the rule's messages, determining if and how to replace the message's default origin-host and origin-realm AVP values with the values configured in the peer profile.

Note: Configuring the Diameter Identity Policy is disabled when the routing rule is defined with either the **Discard** or **Site Proxy** actions.

To set the Diameter Identity Policy:

1. Under Diameter Identity Policy, 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.

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**.

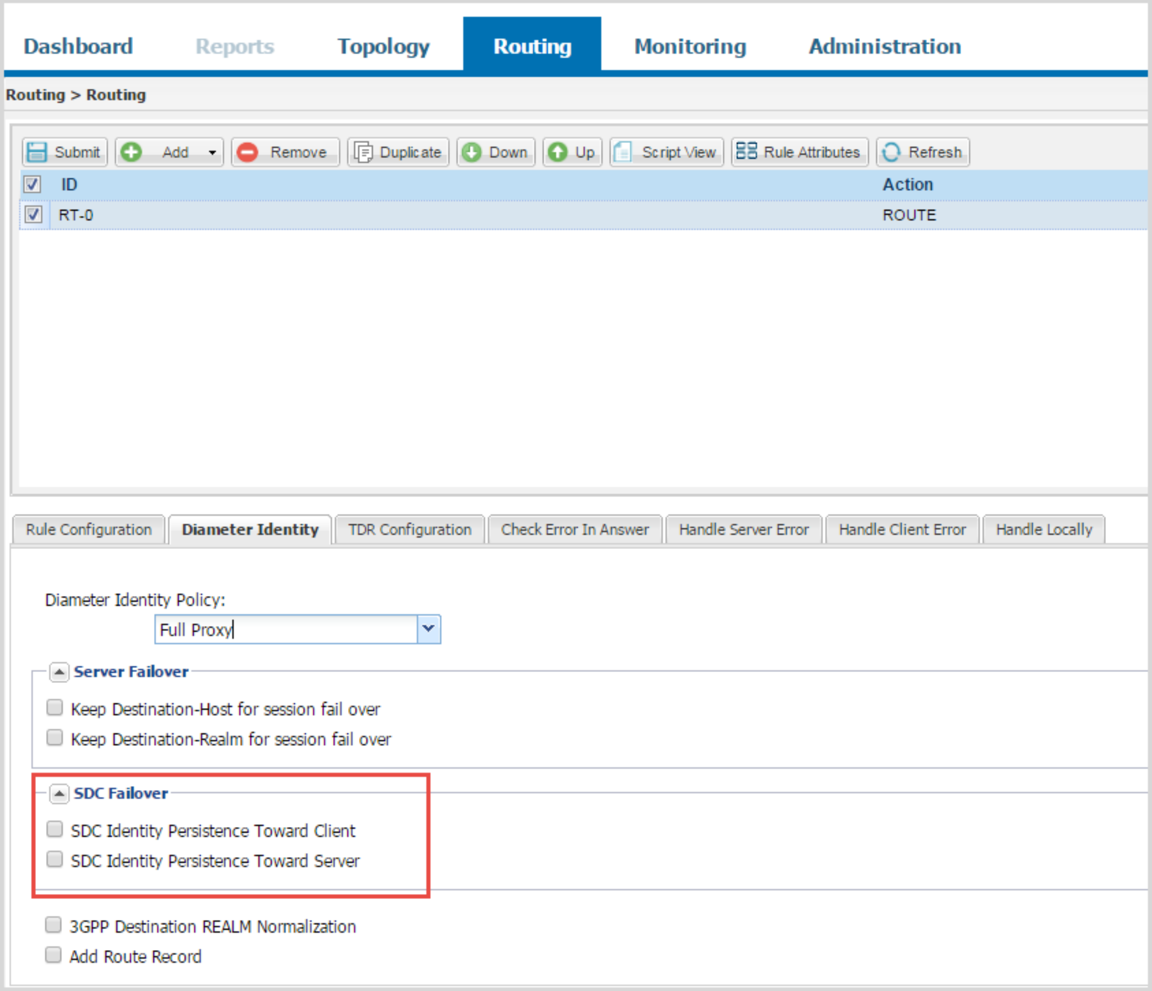
   When selected, the destination-Realm/Host that is sent to the destination server will also be sent to the destination server chosen after a session failover. If not selected, the destination-Realm/Host that will be sent to each destination is the one that was learned during the capabilities exchange.

For all Proxy Diameter Identity policies, (not **Relay**), 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.

To enable the persistence option in an SDC failover:

1. Under **SDC Failover**, select **SDC Identity Persistence Toward Client/Server**.

Figure 44: SDC Diameter Identity Persistence



☑ ID	Action
☑ RT-0	ROUTE

Diameter Identity Policy: Full Proxy

Server Failover

□ Keep Destination-Host for session fail over
□ Keep Destination-Realm for session fail over

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.3gppnet.org for every ULR message.

To add a Route-Record AVP:

1. 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.

4.4.1.2.3 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 tracing for additional AVPs:

1. 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.
3. In the **Value** field, enter the AVP that you want to add to the default set of traced AVPs.

4.4.1.2.4 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.

  Table 34 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:

Table 34: Check Error in Answer Returned Value

Returned Value	Description
RemoteNodeEvent.OK	The answer is transformed to the client.
RemoteNodeEvent.CANNOT_ROUTE	SDC is unable to handle the request. Handle Server Error script is invoked.
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.TIMEOUT	Indicates that no answer was received from the server in the specified time frame. The Request will be resent according to the routing resend parameters.
RemoteNodeEvent.CHANNEL_DISCONNECTED	Indicates that the server did not respond.
RemoteNodeEvent.DNS_PREPARING_POOL	Indicates a DNS preparing failure
RemoteNodeEvent.APPLICATION_ERROR	Indicates an application error.

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 35 shows the **Check Error in Answer script** parameters.

Table 35: 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 35*. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

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

```
resultCode = answer.get("Result-Code").get();
if (resultCode == 4012){
                                                                 return
RemoteNodeEvent.TOO_BUSY;
}
return RemoteNodeEvent.OK;
```

- **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 36 shows the **Handle Server Error** script parameters.

Table 36: Handle Server Error Script Parameters

Handle Server Error Script's Returned Value Type: Message	
Parameter	**Type**
Event	RemoteNodeEvent
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 36*. (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 37 shows the Handle Client Error script parameters.

Table 37: 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 37*. (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 38 details the parameters SDC provides to the script:

Table 38: Handle Locally Script Parameters

Handle Locally Script's Returned Value Type: Boolean	
Parameter	**Type**
Session	Session
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 38*. (To view a detailed list of the SDC Connectivity API methods, click **API** from the menu bar).

Table 39: shows the parameters SDC provides to the script:

Table 39: Handle Locally Script Parameters

Handle Locally Script's Returned Value Type: Message	
Parameter	**Type**
Session	Session
Stack	Stack
incomingMessage	Message
sourceRequest	Message
sourceAnswer	Message

Handle Locally Script’s Returned Value Type: Message	
sourcePeer	Peer
userTraceLogger	UserTraceLoggerWrapper
Metadata	MetaData

- **Redirect**

In **Redirect**, set the script to perform when **Redirect** Routing Action is selected.

Table 40 details the parameters SDC provides to the script:

Table 40: 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 40*. (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 41 shows the parameters SDC provides to the script:

Table 41: 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

Handle Reject Script’s Returned Value Type: 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 **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 42 shows the **Create Message Locally** script parameters.

Table 42: Create Message Locally Script Parameters

Create Answer Locally Script’s Returned Value Type: Message	
Parameter	**Type**
Session	Session
sourceRequest	Message
sourceAnswer	Message
sourcePeer	Peer

Create Answer Locally Script's Returned Value Type: Message	
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 Create Message Locally script:

```
def answer = sourceRequest.createAnswer(2001);
                                                               return answer;
```

4.5 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.

You may also create a criteria-based transformation script, which is operated each time a matching incoming/outgoing message is handled. Transformation scripts define the format by which incoming and outgoing messages are expected to be sent/received to/from the target machine.

The transformation table is divided into two: **Post-Routing** and **Pre-Routing**. Each Transformation rule within the Transformation tables contains a set of parameters, corresponding with the message's content. That is, SDC determines how to transform each message based on its content. Each parameter is represented by a Rule Attribute – an AVP. The Rule attributes are configured independently and each AVP is assigned a type (boolean, regular expression, etc.)

The Rule Attributes list may, for example, consist of the following AVP's: isRequest and DestinationHost. When setting the actual Transformation rules you may use these AVP's to determine the script to invoke when the message is a request **and** its destination host is "Server3" or the action to be taken when the message is not a request. The combination of the Transformation rule's conditions is of type **And**.

Transformation rules are scanned in the order they are listed. The first transformation rule's condition that is met (that is, the message's attributes match the rule's criteria), causes the rule's script invocation.

4.5.1.1 Creating a New Transformation

To create a new Transformation rule:

1. Go to **Routing** > **Transformation**. The Transformation screen is displayed.

The **Post Routing/Pre-Routing** tabs (referring to incoming and outgoing messages, respectively) display the currently empty list of Transformation rules.

Note: You can copy or move up/down an existing Transformation rule, by selecting one of the rows in the table.

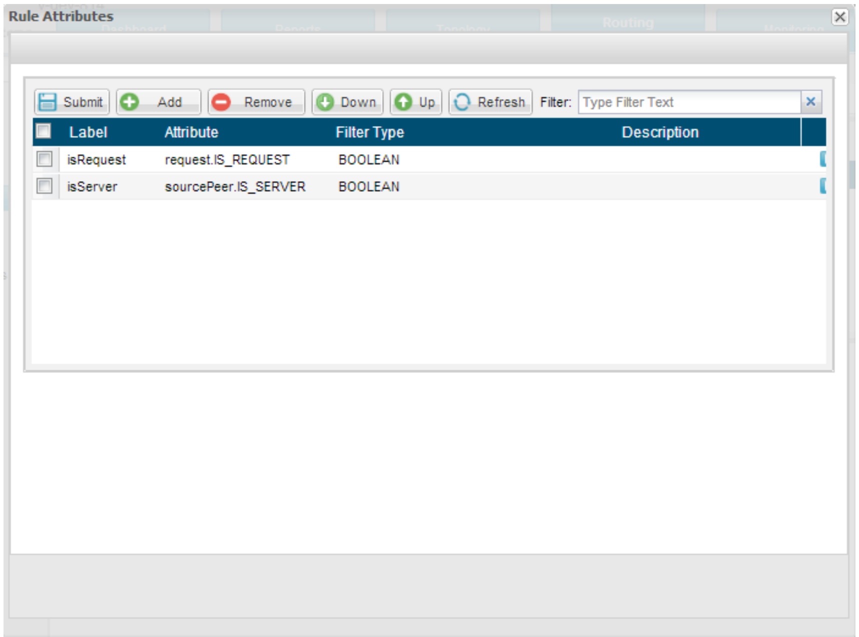
4.5.1.2 Adding a Transformation Rule Attribute

To define the Transformation rules you first need to define their attributes.

To add a transformation rule attribute:

1. Click **Rule Attributes**. The window displays the list of attributes that may be used to define the Transformation rules of Pre-Routing and Post-Routing messages:

Figure 45: Transformation Rule Attributes



□	Label	Attribute	Filter Type	Description
□	isRequest	request.IS_REQUEST	BOOLEAN	
□	isServer	sourcePeer.IS_SERVER	BOOLEAN	

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.: “isRequest”.
4. Under **Attribute**, type in the name of the AVP retrieved from the message. e.g: “request.IS_REQUEST”.
5. Under **Filter Type**, select the data type of the new attribute. e.g.: BOOLEAN. The following figure depicts the available attribute types.
6. Under **Description**, type in a short description of the attribute.
7. Click **Submit**.

Note: For additional information on the decision table attributes, see *Appendix D: Decision Table Attributes*.

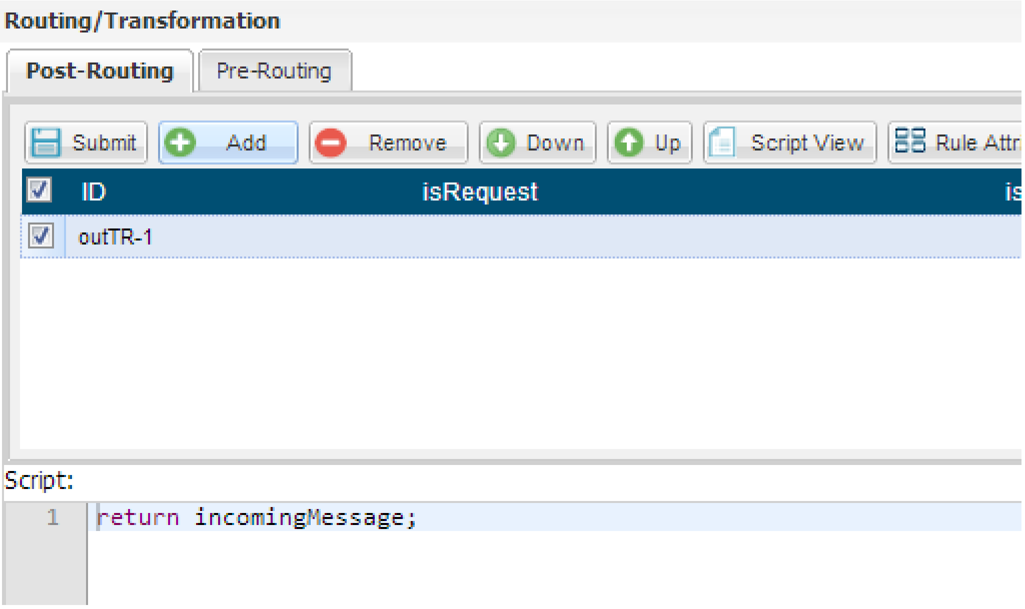
4.5.1.3 Adding a New Transformation Rule

1. Each Transformation Rule, **Pre-Routing** or **Post-Routing**, is associated with the previously defined Rule Attributes. If you have not previously defined any attributes, see *Adding a Transformation Rule Attribute*.

To add a new transformation rule:

1. Click either the **Pre-Routing** tab or **Post-Routing** tab (referring to incoming and outgoing messages, respectively). Each tab displays a table. The table's columns represent the Rule Attributes you previously defined.

Figure 46: Post-Routing Transformation Rules



☑	ID	isRequest	is
☑	outTR-1		

Script:

```
return incomingMessage;
```

2. Click **Add** to create a new Transformation rule. A new rule line assigned an automatic name is added to the table.

3. Under each column, define the value against which new messages are compared. For example: under **isRequest** set the value to “True”, and under **isServer** set the value “to False”.

4. Under **Script**, type in the script to invoke when the conditions of the rule are met.

4.5.1.4 Adding a Transformation Script

You can add a script to be invoked when the conditions of a Transformation rule are met.

Table 43 shows the parameters that SDC provides to the script:

Table 43: 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 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 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;
```

5. Click **Submit**.

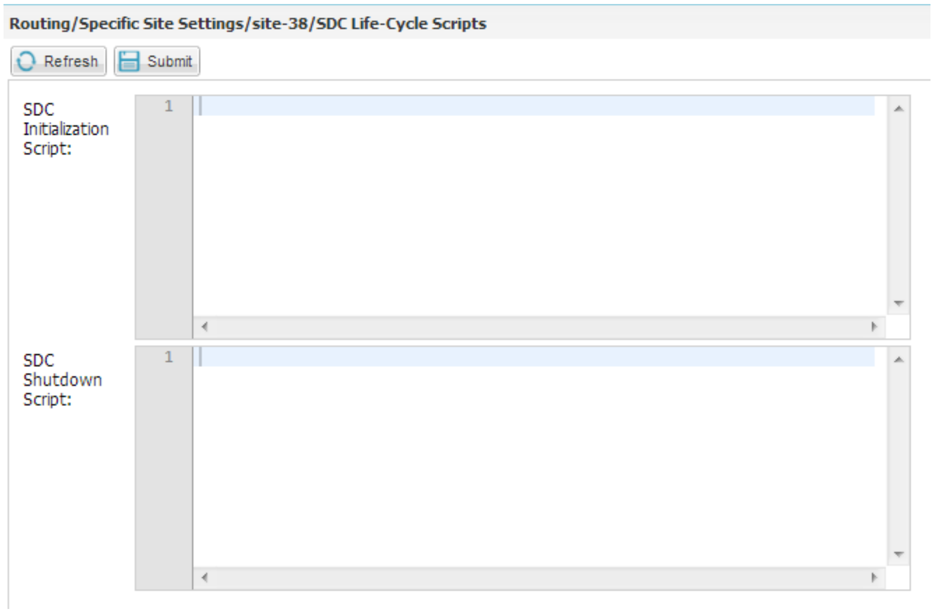
4.6 SDC Life Cycle Scripts

As an SDC Web UI user, you may compose special scripts that run upon each SDC initialization and shutdown. The script 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 **Routing** > **Specific Site Settings** > **SDC Life Cycle Scripts**. The SDC Life Cycle Scripts screen is displayed.

Figure 47: SDC Life Cycle Scripts



Routing/Specific Site Settings/site-38/SDC Life-Cycle Scripts

SDC Initialization Script:

```
1
```

SDC Shutdown Script:

```
1
```

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 44 shows the parameters SDC provides to the scripts:

Table 44: SDC Life Cycle Script Parameters

Parameter	**Type**
Stack	Stack
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).

5. Monitoring the SDC

This chapter describes the different ways that you can monitor the SDC activity and performance. You can view different statistics, traps, and reports from the following Web UI options:

- *SDC Dashboard*
- *EMS Dashboard*
- 🌐 Reports
- *SNMP Traps*
- *Logging and Syslog*
- 🌐 Tracing

In addition, you can view different performance indicators from the **System View**, **System Performance**, and **Host Performance** screens. As part of monitoring SDC activity by viewing traps, you can set **Threshold Management** rates that when they are passed, trigger specific traps.

5.1 🌐 Threshold Management

Threshold Management allows you to set the operational thresholds for alarm execution and KPIs (for additional information on SNMP alarms and KPIs, see *EMS Dashboard*). 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 **System Threshold** categories:

- CPU Utilization
- Disk Utilization
- File system Utilization
- Memory Utilization

- NIC Utilization

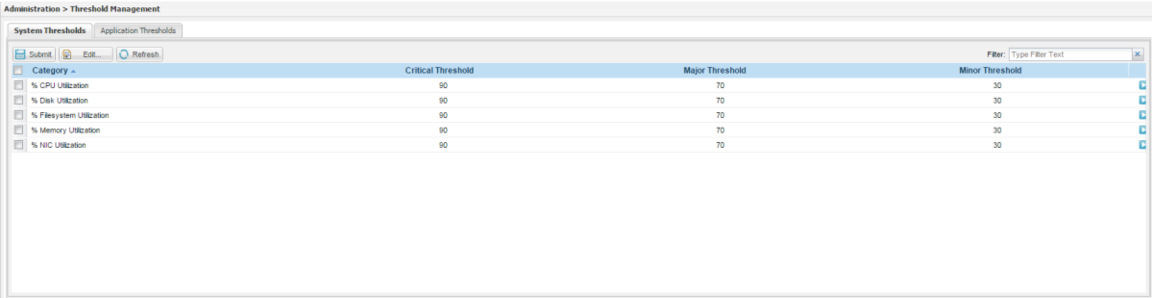
You may also set severity thresholds to the following **Application Threshold** categories:

- Current TPS vs Peer Rate Limit
- Current TPS vs Pool Rate Limit

To set the severity thresholds:

1. Go to **Administration** > **Threshold Management**. The Threshold Management screen is displayed.

Figure 48: Threshold Management



Category	Critical Threshold	Major Threshold	Minor Threshold
% CPU Utilization	90	70	30
% Disk Utilization	90	70	30
% Filesystem Utilization	90	70	30
% Memory Utilization	90	70	30
% NIC Utilization	90	70	30

2. Select the **System** or **Application Threshold** tab, select a category, and then set the **Critical**, **Major** and **Minor** thresholds (i.e., next to CPU Utilization set 70, 50 and 20, respectively).

5.2 SDC Dashboard

SDC Web UI's Dashboard provides a graphical representation of the network flow real-time activities and status, in different time resolutions.

SDC Web UI Statistics provides data concerning:

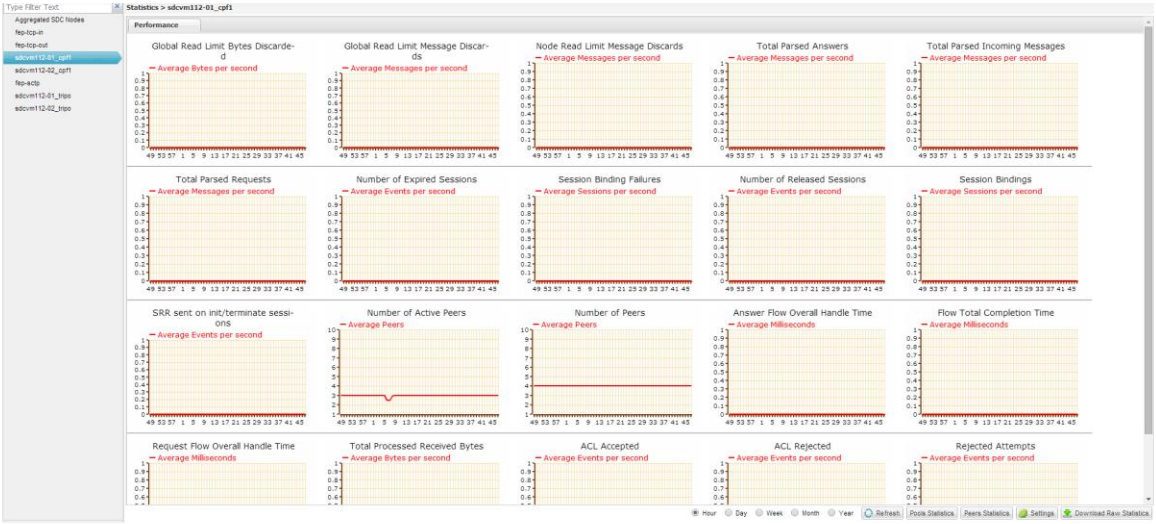
- Network objects (SDC Components, Remote Peers and Pools)
- Flow Management (data flows)

Each of the network objects and logical flows is represented with a designated graph. The graphs' hierarchy is built in accordance with the topology and Flow original hierarchy, so that finding a specific graph is a simple and comfortable drill down.

To view the SDC Dashboard:

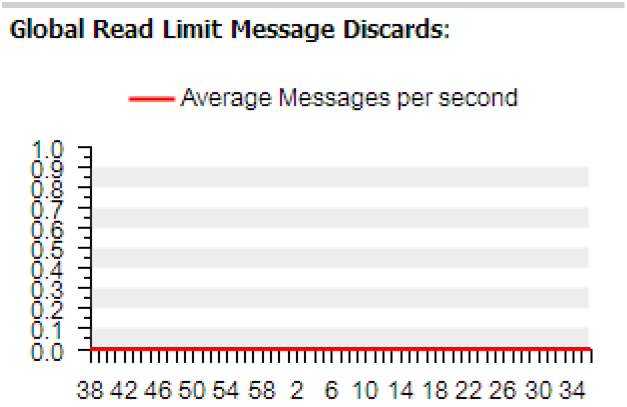
1. Select **Dashboard**.
2. Expand the subfolder pertaining the SDC component that you wish to monitor. The selected reports are displayed. The Collected Data section details the available performance reports.

Figure 49: SDC Dashboard Statistics Graphs


Type Filter Text
Aggregated SDC Nodes
fep-tcp-in
fep-tcp-out
sdcvm112-01_cpf1
sdcvm112-02_cpf1
fep-sctp
sdcvm112-01_tripo
sdcvm112-02_tripo
Statistics > sdcvm112-01_cpf1
Performance
Global Read Limit Bytes Discarde-d
Average Bytes per second
Global Read Limit Message Discar-ds
Average Messages per second
Node Read Limit Message Discards
Average Messages per second
Total Parsed Answers
Average Messages per second
Total Parsed Incoming Messages
Average Messages per second
Total Parsed Requests
Average Messages per second
Number of Expired Sessions
Average Events per second
Session Binding Failures
Average Sessions per second
Number of Released Sessions
Average Events per second
Session Bindings
Average Sessions per second
SRR sent on init/terminate sessi-ons
Average Events per second
Number of Active Peers
Average Peers
Number of Peers
Average Peers
Answer Flow Overall Handle Time
Average Milliseconds
Flow Total Completion Time
Average Milliseconds
Request Flow Overall Handle Time
Average Milliseconds
Total Processed Received Bytes
Average Bytes per second
ACL Accepted
Average Events per second
ACL Rejected
Average Events per second
Rejected Attempts
Average Events per second
49 53 57 1 5 9 13 17 21 25 29 33 37 41 45
● Hour ○ Day ○ Week ○ Month ○ Year
Refresh
Pools Statistics
Peers Statistics
Settings
Download Raw Statistics

A typical line graph is shown in *Figure 50*.

Figure 50: Typical Line Bar



Note: If a graph does not display any data, no activity is currently documented.

3. From the lower right corner of the screen, select the time resolution (hour/day/week/month/year) that the reports display information for and then click **Refresh**. The current report is refreshed to reflect the selected time resolution.

Table 45 describes the available SDC Dashboard graphs.

Table 45: Dashboard Graphs

Report Level	**Report Name**	**Description**	**Data Group**	**Units**
Node	ACL	The number of client connection requests accepted by the SDC based on the Access Control List.	ACL	Events
Node	Answer Flow Overall Handle Time	The time period between T3 and T4 of incoming answers.	Time	Milliseconds
Node	Async Tasks Events Queue Size	The number of requests that are waiting for processing by CPF.	Queue	Events
Node	Async Executor Rejections Events	The number of requests that are not handled (discarded) due to the CPF overload.	Exception	Events
Node	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.	Queue	Events
Node	Messages Executor Rejection Events per Second	The number of incoming message events (requests and	Exception	Events

Report Level	Report Name	Description	Data Group	Units
		answers) rejected by the CPF or FEP.		
Node	Number of Expired Session	The number of expired sessions	Session	Events
Node	Session Binding Failures	The total number of failed session binding attempts.	Session	Sessions
Node	Rejected Attempts	The number of client connection requests rejected by the SDC based on the Access Control List.	ACL	Events
Node	Flow Total Completion Time	The time period between T1 and T4, defined as the total time of a transaction (request and answer).	Time	Milliseconds
Node	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	Bytess
Node	Global Read Limit Message Discard	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	Messages	Messages

Report Level	Report Name	Description	Data Group	Units
		statistic is counted per CPF or FEP.		
Node	Node Read Limit Message Discards	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	Messages
Node	Used Memory	The memory (in bytes) that the CPF and FEP consumed.	System	Bytes
Node	NodeParsedMessages	The average number per second of incoming Diameter and RADIUS messages (requests and answers) that were processed by each CPF per message type.	Messages	Messages
Node	Total Parsed Answers	The total number of answers processed by the CPF or FEP.	Messages	Messages
Node	Total Parsed Incoming Messages	The total number of incoming messages (requests and answers) processed by the CPF or FEP.	Messages	Messages
Node	Total Parsed Requests	The total number of requests processed by the CPF or FEP.	Messages	Messages

Report Level	Report Name	Description	Data Group	Units
Node	Total Processed Received Bytes	The total amount of bytes received and processed by the CPF or FEP.	Bytes	Bytes
Node	Pool 99.95 Percentile of RTT	This presents the pool roundtrip distribution time.	Bytes	Milliseconds
Node	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	Messages
Node	Pool Health	The pool health percentage (between 0% and 100%), based on peer performance in the previous measurement period.	Pool	Percent
Node	Percentage of Timeout Events	The percentage of Timeout Events out of total messages counted per pool.	Pool	Percent
Node	Pool APPLICATION_ERROR Events	The number of APPLICATION_ERROR client pool events	Pool	Messages

Report Level	Report Name	Description	Data Group	Units
Node	Pool Overloaded Events	The number of OVERLOAD client pool events	Pool	Messages
Node	Pool Ramp-Up Overloaded Events	The number of RAMPUP_OVERLOADED client pool events	Pool	Messages
Node	Pool TIMEOUT Events	The number of TIMEOUT client pool events	Pool	Messages
Node	Pool TOO_BUSY Events	The number of TOO_BUSY client pool events	Pool	Messages
Node	Pool Average Roundtrip Time	The pool roundtrip time of messages routed using the pool.	Pool	Millisecondssss
Node	Pool Sent Messages	The number of sent messages per pool.	Pool	Messages
Node	Pool Total Answers Received	The number of answers received per pool.	Pool	Messages
Node	Answer Flow Handle Time (by Protocol)	The time period between T3 and T4 of incoming answers, per protocol.	Time	Milliseconds
Node	Request Flow Handle Time (by Protocol)	The time period between T2 and T1 of incoming requests, per protocol.	Time	Milliseconds
Node	Session Releases	The number of session that were released.	Session	Events

Report Level	Report Name	Description	Data Group	Units
Node	Request Flow Overall Handle Time	The time period between T2 and T1 of incoming requests.	Time	Milliseconds
Node	Session Bindings	The total number of successful session binding attempts.	Session	Sessions
Node	Decision Table	The number of requests handled by a routing/transformation/ session management rule.	Decision Tables	Messages
Node	SRR sent on init/terminate sessions	The number of SRRs sent to the mated SDC site for session initiations and session terminations.	Session	Events
Node	Number of Active Peers	The number of open peers connected to the CPF.	Peer	Peers
Node	Number of Peers	The number of peers connected to the CPF.	Peer	Peers
Node	Failed send attempts of SRRs during full SDC site replication	Counts the number of acknowledged/failed/expired SRRs sent during full SDC site replication. The statistic is counted per Tripo instance that sends the SRR.	Messages	Messages

Report Level	Report Name	Description	Data Group	Units
Node	Failed send attempts of SRRs during re-synchronization	Counts the number of acknowledged/ failed/expired SRRs sent during re-synchronization of the replication queue. The statistic is counted per Tripo instance that sends the SRR.	Messages	Messages
Node	Failed Received SRR attempts	Counts the number of received SRRs successful/failed attempts. The statistic is counted per Tripo instance that receives the SRR.	Messages	Messages
Node	Failed send attempts of SRRs	Counts the number of acknowledged/failed/expired SRRs sent to the Mated SDC site. The statistic is counted per Tripo instance that sends the SRR.	Messages	Messages
Node	Successfully sent SRRs during full SDC site replication	Counts the number of acknowledged/failed/expired SRRs sent during full SDC site replication. The statistic is counted per Tripo instance that sends the SRR.	Messages	Messages

Report Level	Report Name	Description	Data Group	Units
Node	Successfully sent SRRs during re-synchronization	Counts 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.	Messages	Messages
Node	Received SRRs	Counts the number of received SRRs successful/failed attempts. The statistic is counted per Tripo instance that receives the SRR.	Messages	Messages
Node	Replication Sent Success	Counts the number of acknowledged/failed/expired SRRs sent to the Mated SDC site. The statistic is counted per Tripo instance that sendss the SRR.	Messages	Messages
Peer	Discarded Messages (by Message Type)	The number of discarded messages (per message type) due to channel disconnections between the FEP and CPF.	System	Messages
Peer	Peer Local Read Limit Message Discard	The number of discarded messages due	Messages	Messages

Report Level	Report Name	Description	Data Group	Units
		to the configured read rate limit per origin peer. This statistic is counted per origin peer.		
Peer	Peer Read Limit Message Discards	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	Messages
Peer	Peer Effective Capacity	The projected peer capacity, based on the combination of the configured rate limit and the real capacity measured in the	Messages	Messages

Report Level	Report Name	Description	Data Group	Units
		previous measurement period.		
Peer	Peer Health	The peer health percentage (between 0% and 100%), based on peer performance in the previous measurement period.	Messages	Percent
Peer	Sent Messages	The average number of messages sent, counted per destination peer.	Messages	Messages
Peer	Pending Requests	The average number of requests waiting for an answer per destination peer.	Peer	Messages
Peer	Received Bytes	The amount of bytes received, counted per origin peer.	Bytes	Bytes
Peer	Received Messages (by Message Type)	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	Messages

Report Level	Report Name	Description	Data Group	Units
Peer	Received Messages Before Read Discard	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	Messages
Peer	peer APPLICATION_ERROR events	The number of APPLICATION_ERROR client peer events	Bytes	Messages
Peer	peer CANNOT_ROUTE events	The number of CANNOT_ROUTE client peer events	Bytes	Messages
Peer	peer CHANNEL_DISCONNECTED events	The number of CHANNEL_DISCONNECTED client peer events	Bytes	Messages
Peer	peer DNS_PREPARING_ POOL events	The number of DNS_PREPARING_POOL client peer events	Bytes	Messages
Peer	Peer OK Events	The number of OK client peer events	Bytes	Messages
Peer	peer REDIRECT events	The number of REDIRECT client peer events	Bytes	Messages

Report Level	Report Name	Description	Data Group	Units
Peer	Peer REQUEST_REJECTED Events	The number of REQUEST_REJECTED client peer events	Bytes	Messages
Peer	peer_TIMEOUT events	The number of TIMEOUT client peer events	Bytes	Messages
Peer	peer TOO_BUSY events	The number of TOO_BUSY client peer events	Bytes	Messages
Peer	RoundtripT ime	The average time (in milliseconds), of request processing by the destination (T3-T2), counted per source peer and message type.	Time	Milliseconds
Peer	Peer Average Roundtrip Time	The time period between T2 and T3, defined as the request processing time by destination.	Time	Milliseconds
Peer	Peer Percentile 99.95% Roundtrip Time	This presents 99.95% of the destination peer latency (T3-T2).	Time	Milliseconds
Peer	Sent Bytes	The amount of bytes sent, counted per destination peer.	Bytes	Bytes
Peer	Sent Messages (by Message Type)	The average number of messages routed by the CPF per destination	Messages	Messages

Report Level	Report Name	Description	Data Group	Units
		peer. Counted by message type.		
Peer	Retransmission Timeout Events per Server	The number of requests that were retransmitted, counted per destination peer and message type. (Counted for RADIUS messages only).	Exception	Messages
Peer	peer TIMEOUT Events	The number of unanswered requests due to timeout, per destination peer and per message type.	Exception	Messages
Routing	DNS Resolving Succeeded	DNS Resolving Succeeded to Get the Destination	Peer	Rules
Routing	DNS Resolving Failed	DNS Resolving Failed to Get the Destination	Exception	Rules

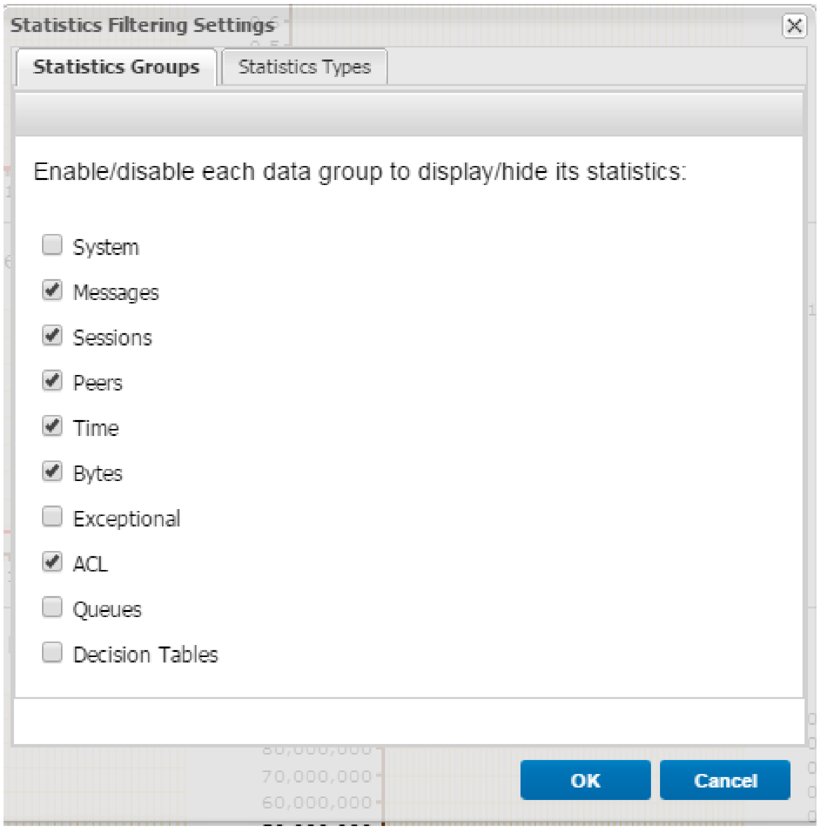
5.2.1 Statistics Filtering Settings

The Statistics Settings allow you to filter the type of graphs you wish to display in the SDC Dashboard. Each statistics type belongs to a statistics group. That is, if you wish to display or hide all members of a statistics group, use the Statistics Groups tab. For instance, if you wish to hide the parameters “Pending Request”, “Number of Peers” and “Number of Active Peers”, you do not have to individually add each of them to the Statistics Types table. You may, alternatively, simply uncheck “Peers” under Statistics group. However, if you wish to display a few group members and hide others, use the Statistics Types tab.

To filter the type of statistics data displayed in the SDC Dashboard:

1. From the bottom right corner of the screen, click **Settings**.

Figure 51: Statistics Filtering Settings



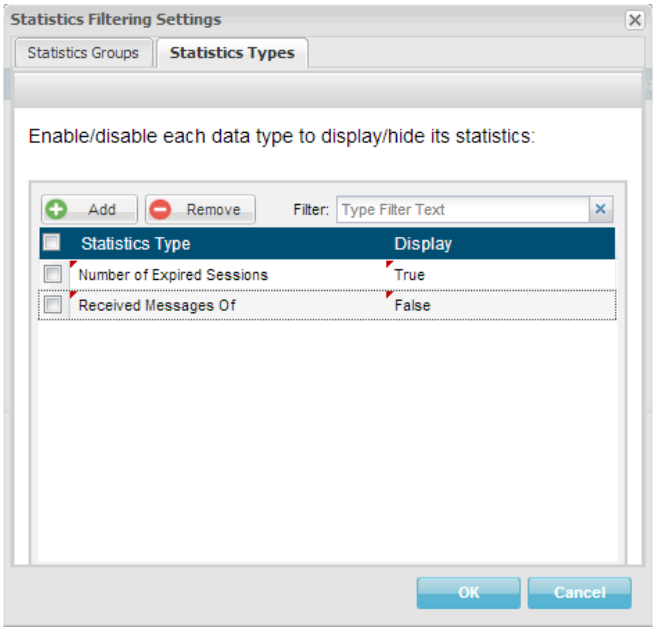
Enable/disable each data group to display/hide its statistics:

☐ System
☑ Messages
☑ Sessions
☑ Peers
☑ Time
☑ Bytes
☐ Exceptional
☑ ACL
☐ Queues
☐ Decision Tables

2. Under **Statistics Groups**, select/clear each of the data groups that you want to display or hide the statistics for. If, for example, you want to hide all message related graphs, clear the “Messages” checkbox.

3. Under **Statistics Types**, click **Add** to add a specific data type of which you wish to display or hide the statistics and under **Display** column, select True or False respectively. If, for example, you wish to hide the graphs of the parameter “**Number of Expired Sessions**”, add it to the Statistics Types table and select **False** under **Display**, as shown in the following image:

Figure 52: Statistics Types



Enable/disable each data type to display/hide its statistics:

□	Statistics Type	Display
□	Number of Expired Sessions	True
□	Received Messages Of	False

Note: **Statistics Types** properties are stronger than **Statistics Groups** properties. That is, if you select a statistics group to be displayed in SDC Web UI and set one of its members' entries to **False**, the entry's graph will not be displayed. However, if you remove an entry from the Statistics Types while the group to which the entry belongs is still selected in the **Statistics Groups** tab, the graph of the entry you removed will be displayed. For example: if you choose to remove "Received Messages" from the **Statistics Types** tab (assigned with either True or False Display property) and "Messages" group is still selected in the **Statistics Groups** tab, the Received Messages graph will be displayed despite removing it from the list.

5.2.2 Statistic Data Collection

The SDC collects various types of statistic records. The data is collected, analyzed and stores it in a designated output file. The SDC also collects the following time stamps that reflect each new transaction's roundtrip:

Table 46: Transaction's Roundtrip Time Stamps

Time Stamp	Description
T1	The time when SDC received the request from the Client Remote Node
T2	The time when SDC sent the request to the Server Remote Node
T3	The time when SDC received the answer from the Server Remote Node
T4	The time when SDC sent the answer to the Client Remote Node

SDC uses the time stamps described in *Table 46* to calculate the following data:

- Flow Total Completion Time (the difference between T4 and T1)
- Peer Message Processing Time Average (the difference between T4 and T3/T2 and T1)
- Peer Roundtrip Time Average (the difference between T3 and T2)

5.2.3 Configuring the Data Collection

Two parameters are used to set the statistic data collection and the display format in the output file. These parameters are configured through the defaultLBConfiguration.xml file.

- RawStatisticLogFile – this parameter sets the name and path of the collected data output file. The parameter is a string formatted parameter.
- RawStatisticOneLineFormat – the parameter determines whether the output file displays data in one line or in two lines. The parameter is Boolean formatted parameter, and the default value is set to false.
    - False – the file displays data in a two line format: the upper line displays the statistic type name and the value is displayed in bottom line, delimited by semicolon.
    - True – the file displays data in one line according to the following order: statistic type name, space, value, semicolon, the following statistic type name, space, value.

5.3 EMS Dashboard

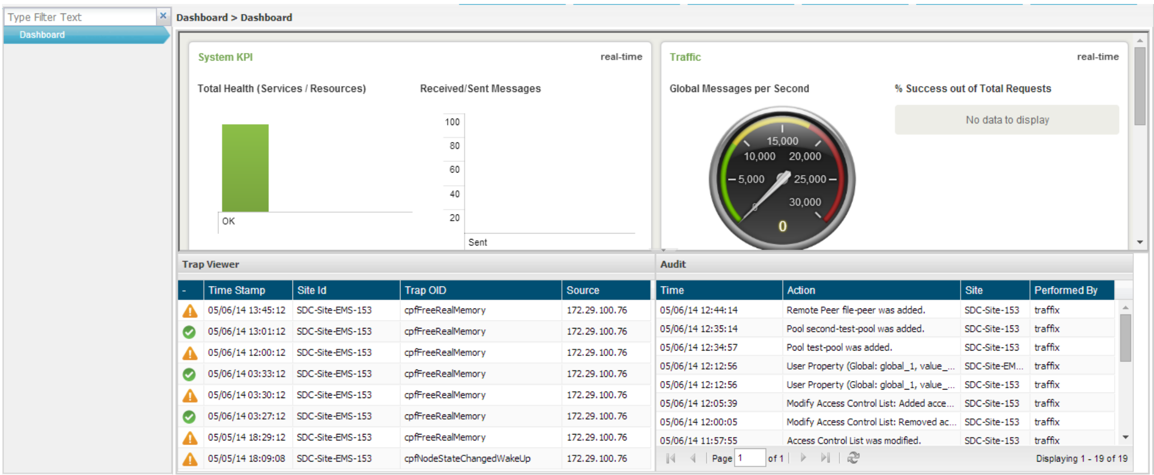
The EMS Dashboard provides a central display of main real-time key performance indicators, statistics graphs and recently generated SNMP traps.

To view the EMS Dashboard:

1. From the tab menu click **Dashboard**.

   The Dashboard screen is displayed, as depicted in the following image:

Figure 53: EMS Dashboard Display



Trap Viewer

-	Time Stamp	Site Id	Trap OID	Source
⚠	05/06/14 13:45:12	SDC-Site-EMS-153	cpfFreeRealMemory	172.29.100.76
✓	05/06/14 13:01:12	SDC-Site-EMS-153	cpfFreeRealMemory	172.29.100.76
⚠	05/06/14 12:00:12	SDC-Site-EMS-153	cpfFreeRealMemory	172.29.100.76
✓	05/06/14 03:33:12	SDC-Site-EMS-153	cpfFreeRealMemory	172.29.100.76
⚠	05/06/14 03:30:12	SDC-Site-EMS-153	cpfFreeRealMemory	172.29.100.76
✓	05/06/14 03:27:12	SDC-Site-EMS-153	cpfFreeRealMemory	172.29.100.76
⚠	05/05/14 18:29:12	SDC-Site-EMS-153	cpfFreeRealMemory	172.29.100.76
⚠	05/05/14 18:09:08	SDC-Site-EMS-153	cpfNodeStateChangedWakeUp	172.29.100.76

Audit

Time	Action	Site	Performed By
05/06/14 12:44:14	Remote Peer file-peer was added.	SDC-Site-153	traffix
05/06/14 12:35:14	Pool second-test-pool was added.	SDC-Site-153	traffix
05/06/14 12:34:57	Pool test-pool was added.	SDC-Site-153	traffix
05/06/14 12:12:56	User Property (Global: global_1, value_...	SDC-Site-EM...	traffix
05/06/14 12:12:56	User Property (Global: global_1, value_...	SDC-Site-153	traffix
05/06/14 12:05:39	Modify Access Control List: Added acce...	SDC-Site-153	traffix
05/06/14 12:00:05	Modify Access Control List: Removed ac...	SDC-Site-153	traffix
05/06/14 11:57:55	Access Control List was modified.	SDC-Site-153	traffix

Page 1 of 1 Displaying 1 - 19 of 19

Table 47 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 47: 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 and working)/Warning (the service/resource was marked as failed at least once in the last minute)/Critical (the

Graph	Description
	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.

5.4 Reports

The Reports tab provides you with statistics data gathered by EMS and displayed according to your preference.

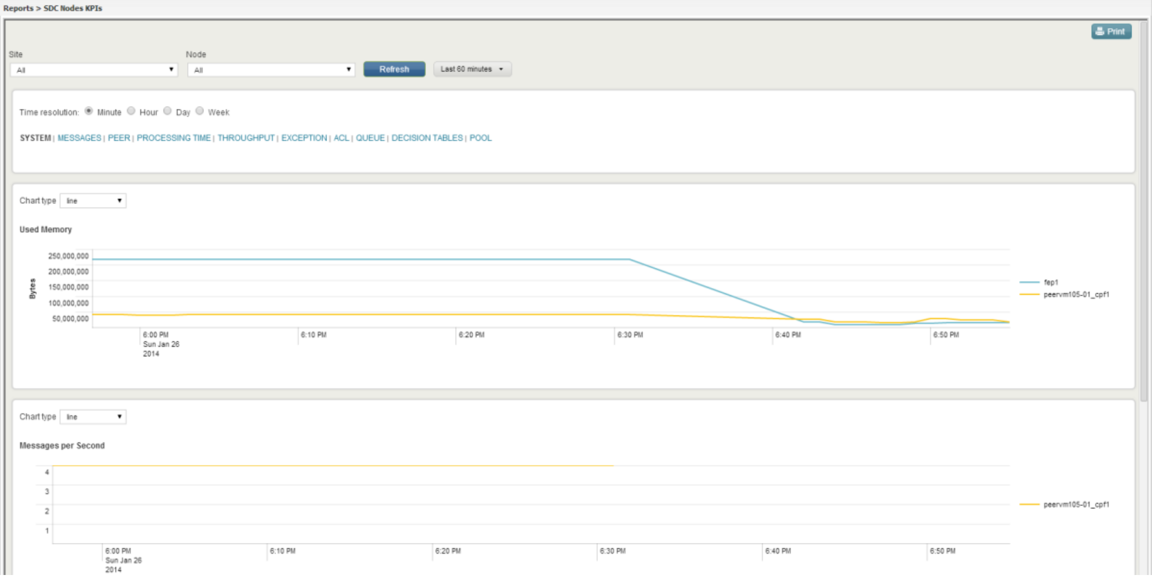
5.4.1 SDC Node KPIs

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 54: SDC Node KPIs Reports



Reports > SDC Nodes KPIs

Site: All

Node: All

Last 60 minutes

Time resolution: ● Minute ○ Hour ○ Day ○ Week

SYSTEM | MESSAGES | PEER | PROCESSING TIME | THROUGHPUT | EXCEPTION | ACL | QUEUE | DECISION TABLES | POOL

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 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 |ACL |QUEUE |DECISION TABLES |POOL

6. Select the **Chart Type**.

Table 48 details the available report types:

Table 48: SDC Node KPI Report Types

Category	Report	Description
System	Used Memory	The memory (in bytes) 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 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.

Category	Report	Description
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-In
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.
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.

Category	Report	Description
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.
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

Category	Report	Description
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 Recieved per Second	Number of received messages per pool

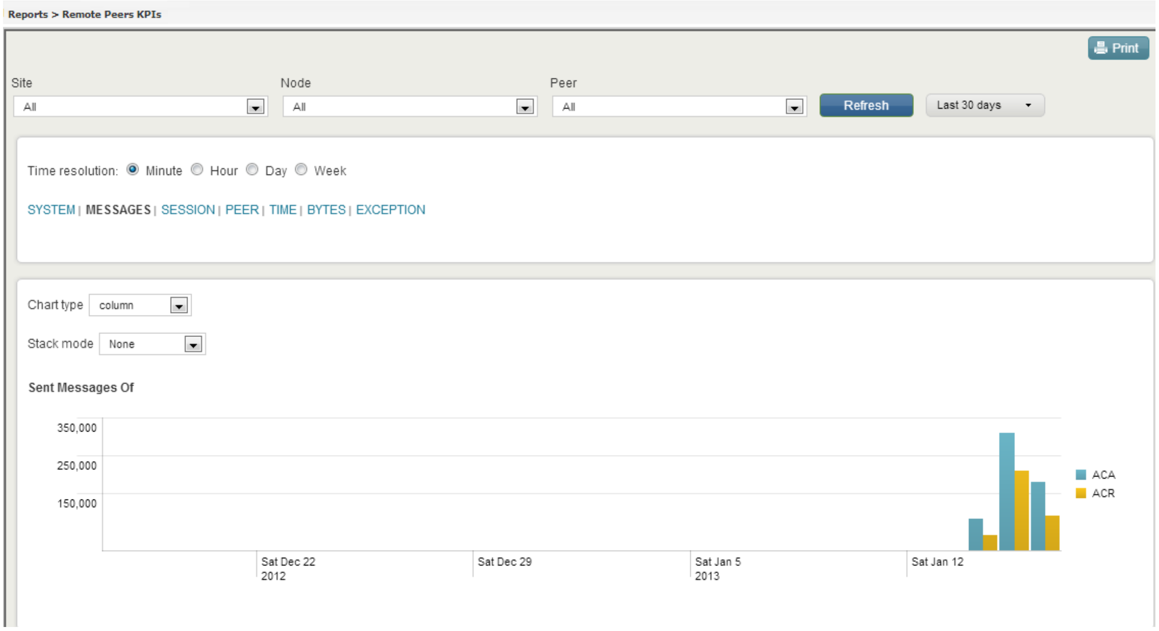
5.4.2 Remote Peer KPIs

The Remote Peer KPI reports display the number of sent and failed messages per client per message type per error event.

To view a remote peer KPI report:

1. Go to **Reports > Remote Peer KPIs**. The Remote Peer KPIs screen is displayed.

Figure 55: Remote Peer KPI Reports



Reports > Remote Peers KPIs

Site: All
Node: All
Peer: All
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 49 details the available report types:

Table 49: 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

Category	Report	Description
		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.
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.

Category	Report	Description
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.
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.

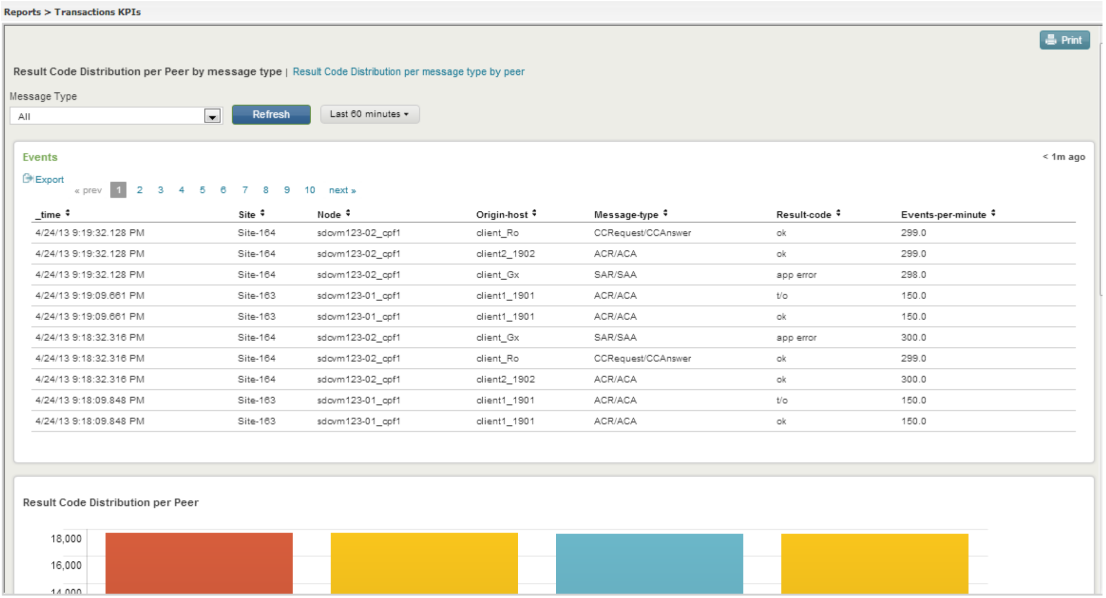
5.4.3 Transactions KPIs

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 56: 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 50*.

Table 50: 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

Result Code	Remote Node Events
	▪ PeerRemoteNodeEvents_CHANNEL_DISCONNECTED ▪ PeerRemoteNodeEvents_REQUEST_REJECTED ▪ PeerRemoteNodeEvents_REDIRECT ▪ PeerRemoteNodeEvents_APPLICATION_ERROR

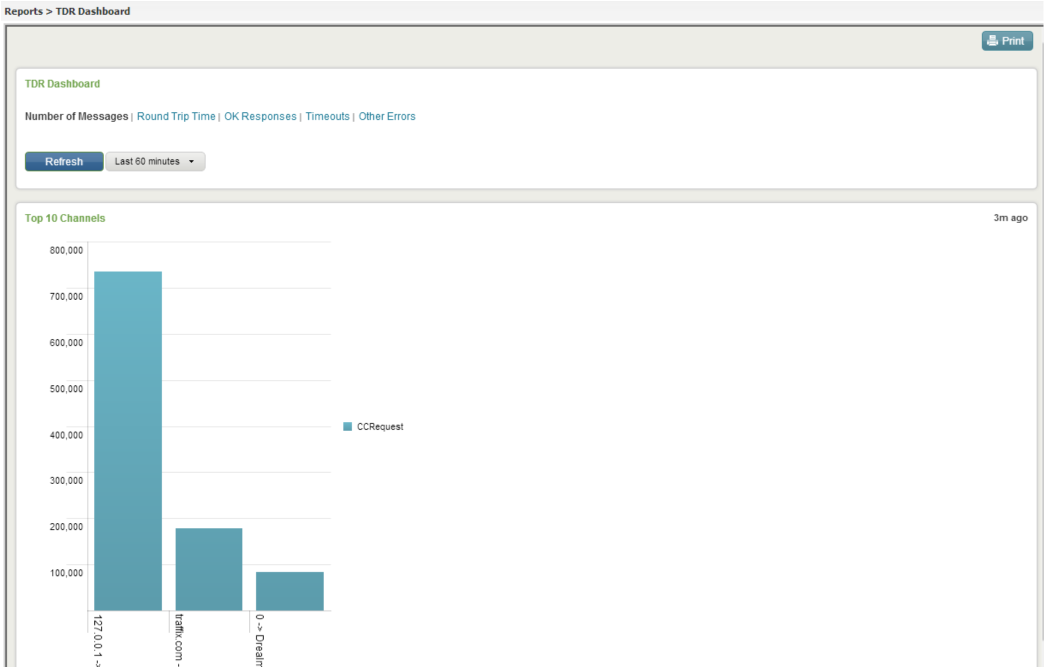
5.4.4 TDR Dashboard

The TDR Dashboard (shown in *Figure 57*) 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.

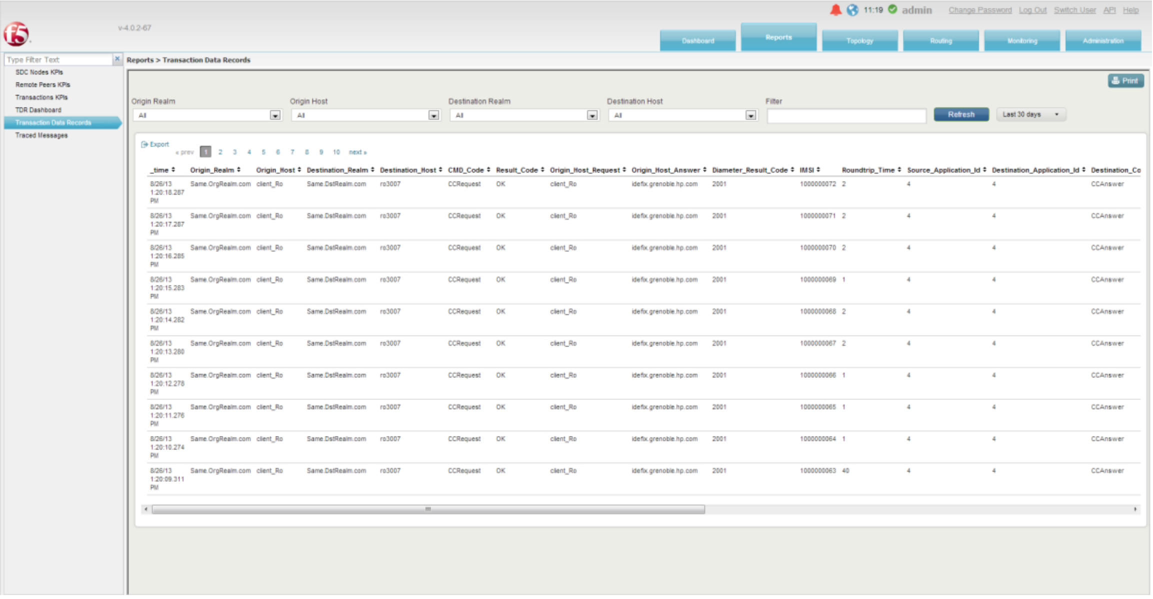
Figure 57: Reports>TDR Dashboard – Number of Messages view



5.4.5 Transaction 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. *Figure 58* is an example of the Transaction Data Reports screen with no filters applied to the displayed data.

Figure 58: Reports>Transaction Data Records



Origin Realm: All

Origin Host: All

Destination Realm: All

Destination Host: All

_time	Origin_Realm	Origin_Host	Destination_Realm	Destination_Host	CMD_Code	Result_Code	Origin_Host_Request	Origin_Host_Answer	Diameter_Result_Code	IMSI	Roundtrip_Time	Source_Application_Id	Destination_Application_Id	Destination_Co
8/26/13 1:20:18.287 PM	Same.OrgRealm.com	client_Ro	Same.DstRealm.com	ro3007	CCRequest	OK	client_Ro	idefix.grenoble.hp.com	2001	100000072	2	4	4	CCAnswer
8/26/13 1:20:17.287 PM	Same.OrgRealm.com	client_Ro	Same.DstRealm.com	ro3007	CCRequest	OK	client_Ro	idefix.grenoble.hp.com	2001	100000071	2	4	4	CCAnswer
8/26/13 1:20:16.285 PM	Same.OrgRealm.com	client_Ro	Same.DstRealm.com	ro3007	CCRequest	OK	client_Ro	idefix.grenoble.hp.com	2001	100000070	2	4	4	CCAnswer
8/26/13 1:20:15.283 PM	Same.OrgRealm.com	client_Ro	Same.DstRealm.com	ro3007	CCRequest	OK	client_Ro	idefix.grenoble.hp.com	2001	100000069	1	4	4	CCAnswer
8/26/13 1:20:14.282 PM	Same.OrgRealm.com	client_Ro	Same.DstRealm.com	ro3007	CCRequest	OK	client_Ro	idefix.grenoble.hp.com	2001	100000068	2	4	4	CCAnswer
8/26/13 1:20:13.280 PM	Same.OrgRealm.com	client_Ro	Same.DstRealm.com	ro3007	CCRequest	OK	client_Ro	idefix.grenoble.hp.com	2001	100000067	2	4	4	CCAnswer
8/26/13 1:20:12.278 PM	Same.OrgRealm.com	client_Ro	Same.DstRealm.com	ro3007	CCRequest	OK	client_Ro	idefix.grenoble.hp.com	2001	100000066	1	4	4	CCAnswer
8/26/13 1:20:11.276 PM	Same.OrgRealm.com	client_Ro	Same.DstRealm.com	ro3007	CCRequest	OK	client_Ro	idefix.grenoble.hp.com	2001	100000065	1	4	4	CCAnswer
8/26/13 1:20:10.274 PM	Same.OrgRealm.com	client_Ro	Same.DstRealm.com	ro3007	CCRequest	OK	client_Ro	idefix.grenoble.hp.com	2001	100000064	1	4	4	CCAnswer
8/26/13 1:20:09.311 PM	Same.OrgRealm.com	client_Ro	Same.DstRealm.com	ro3007	CCRequest	OK	client_Ro	idefix.grenoble.hp.com	2001	100000063	40	4	4	CCAnswer

Table 51 details the collected data in each generated TDR.

Table 51: 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.

Data Field	Data Type	Description
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.
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.

Data Field	Data Type	Description
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.

5.4.6 Traced Messages

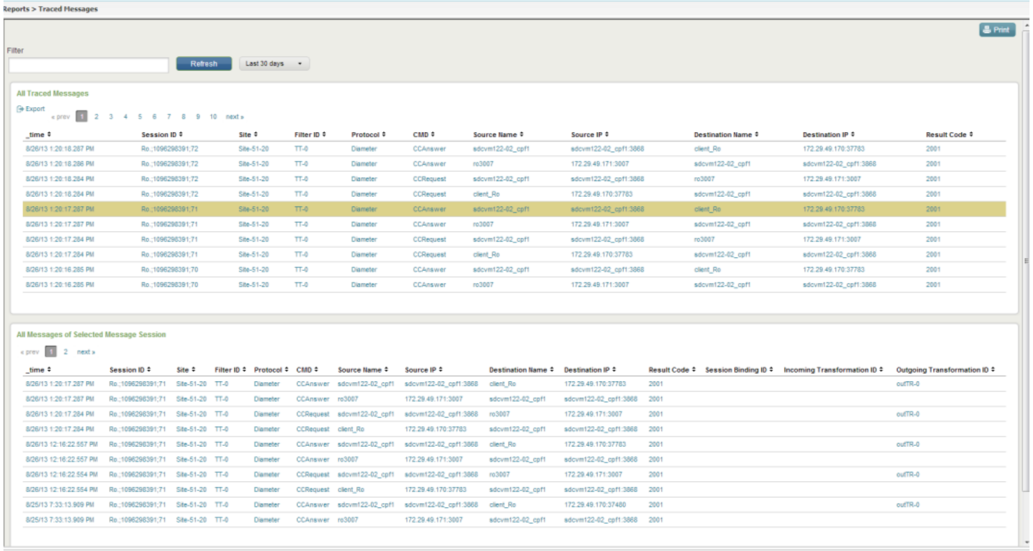
The Traced Messages displays a log of transactions made in your system.

Note: To activate message tracing see *Configuring a Tracing Rule.*

To view traced messages:

1. Go to **Reports** > **Traced Messages**. The **Traced Messages** screen is displayed.

Figure 59: 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

_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.

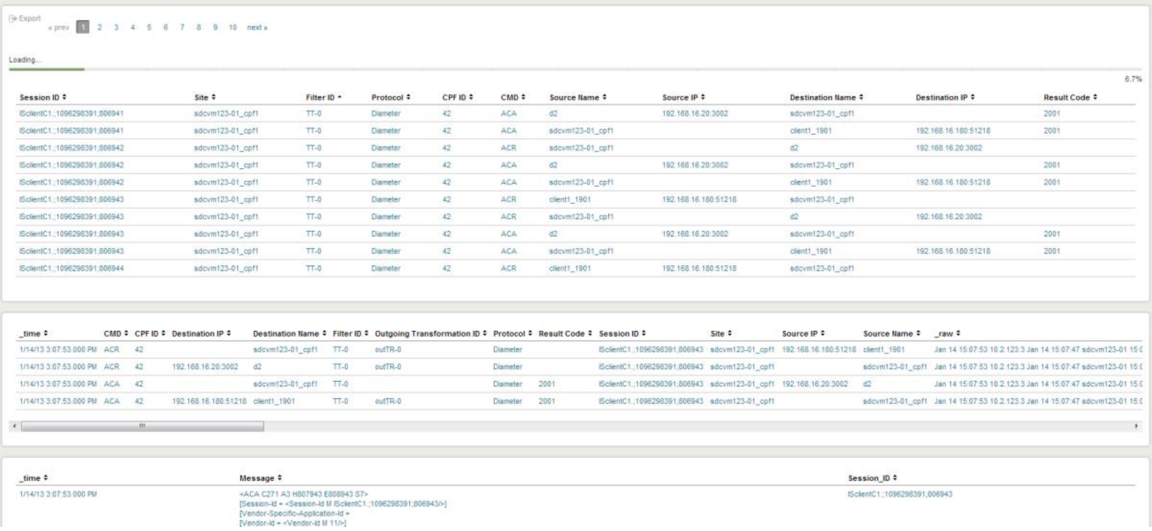
Each transaction is comprised of four messages:

- 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, as shown in *Figure 60*.

Figure 60: Traced Messages – 4 Messages



Session ID ⇕	Site ⇕	Filter ID ▴	Protocol ⇕	CPF ID ⇕	CMD ⇕	Source Name ⇕	Source IP ⇕	Destination Name ⇕	Destination IP ⇕	Result Code ⇕
lSclientC1.;1096298391;806941	sdcvm123-01_cpf1	TT-0	Diameter	42	ACA	d2	192.168.16.20:3002	sdcvm123-01_cpf1		2001
lSclientC1.;1096298391;806941	sdcvm123-01_cpf1	TT-0	Diameter	42	ACA	sdcvm123-01_cpf1		client1_1901	192.168.16.180:51218	2001
lSclientC1.;1096298391;806942	sdcvm123-01_cpf1	TT-0	Diameter	42	ACR	sdcvm123-01_cpf1		d2	192.168.16.20:3002	
lSclientC1.;1096298391;806942	sdcvm123-01_cpf1	TT-0	Diameter	42	ACA	d2	192.168.16.20:3002	sdcvm123-01_cpf1		2001
lSclientC1.;1096298391;806942	sdcvm123-01_cpf1	TT-0	Diameter	42	ACA	sdcvm123-01_cpf1		client1_1901	192.168.16.180:51218	2001
lSclientC1.;1096298391;806943	sdcvm123-01_cpf1	TT-0	Diameter	42	ACR	client1_1901	192.168.16.180:51218	sdcvm123-01_cpf1		
lSclientC1.;1096298391;806943	sdcvm123-01_cpf1	TT-0	Diameter	42	ACR	sdcvm123-01_cpf1		d2	192.168.16.20:3002	
lSclientC1.;1096298391;806943	sdcvm123-01_cpf1	TT-0	Diameter	42	ACA	d2	192.168.16.20:3002	sdcvm123-01_cpf1		2001
lSclientC1.;1096298391;806943	sdcvm123-01_cpf1	TT-0	Diameter	42	ACA	sdcvm123-01_cpf1		client1_1901	192.168.16.180:51218	2001
lSclientC1.;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		lSclientC1.;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 15:0
1/14/13 3:07:53.000 PM	ACR	42	192.168.16.20:3002	d2	TT-0	outTR-0	Diameter		lSclientC1.;1096298391;806943	sdcvm123-01_cpf1		sdcvm123-01_cpf1	Jan 14 15:07:53 10.2.123.3 Jan 14 15:07:47 sdcvm123-01 15:0
1/14/13 3:07:53.000 PM	ACA	42		sdcvm123-01_cpf1	TT-0		Diameter	2001	lSclientC1.;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 15:0
1/14/13 3:07:53.000 PM	ACA	42	192.168.16.180:51218	client1_1901	TT-0	outTR-0	Diameter	2001	lSclientC1.;1096298391;806943	sdcvm123-01_cpf1		sdcvm123-01_cpf1	Jan 14 15:07:53 10.2.123.3 Jan 14 15:07:47 sdcvm123-01 15:0

_time ⇕	Message ⇕	Session_ID ⇕
1/14/13 3:07:53.000 PM	<ACA C271 A3 H807943 E808943 S7> [Session-Id = <Session-Id M lSclientC1.;1096298391;806943/>] [Vendor-Specific-Application-Id = [Vendor-Id = <Vendor-Id M 11/>]	lSclientC1.;1096298391;806943

5.4.7 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 52 details the available report types.

Table 52: 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.
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.

Category	Report	Description
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.
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.

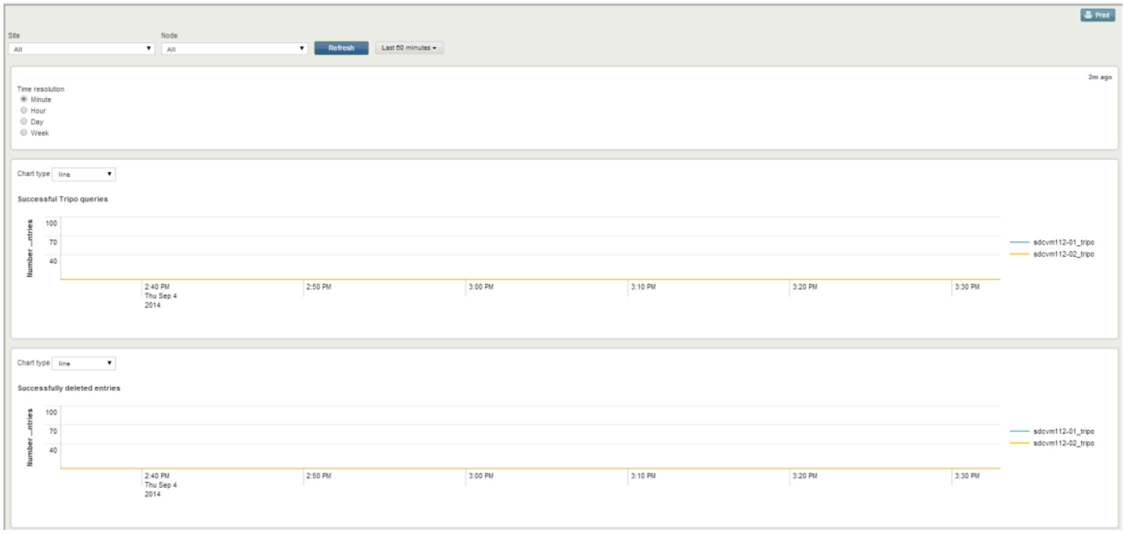
5.4.8 Repository KPIs

The Repository KPIs reports display information about sessions saved in the Tripo.

To view a repository KPI report:

1. Go to **Reports** > **Repository KPIs**. The Repository KPIs screen is displayed.

Figure 61: Repository KPIs Reports



Site: All

Node: All

Refresh

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 53 details the available report types.

Table 53: 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.

Report	Description
Failed addition attempts (Tripo overflow)	The number of failed additional Tripo attempts as a result of a Tripo storage overflow.
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

5.5 SNMP Traps

SDC's monitoring and fault analysis is 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 Defining the SNMP Target

As an SDC manager, you can change the target machines to which SNMP traps are sent upon execution and prevent the alarms from flooding the system.

SNMP Targets define where SDC's traps are sent upon execution. Traps invoked by the local site are sent to the configured targets and to the NMS manager. The NMS manager session maps the traps to the configured targets on the manager site.

To set the SDC trap targets:

1. Go to **Administration** > **Specific Site Settings** > **SNMP > SNMP Targets**.

   The **SNMP Targets** screen is displayed.

Figure 62: SNMP Targets

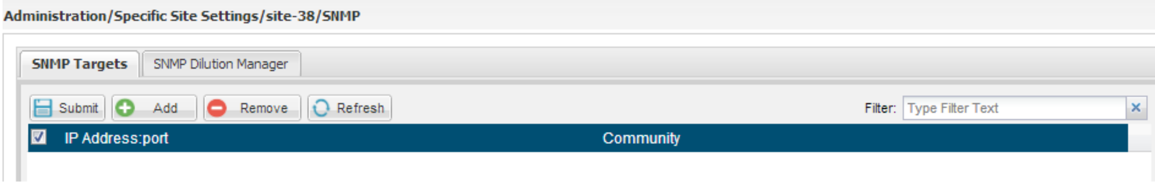


Table 54 details the SNMP Target table properties:

Table 54: SNMP Targets Table

Column	Description
IP Address port	The IP address to which Alarms are sent upon execution.
Community	Defines the SNMP community that is used by the SDC to publish its trap. It must correlates with the trap Target's community. SDC's community string is set to 'public' by default, and you may change it.

5.5.2 SNMP Dilution Manager

To prevent the alarms from flooding the system, SDC provides a dilution and filtering mechanism. Each alarm is assigned a maximum event occurrence number in a specified measuring interval, after which a dilution period, in which no traps are invoked, begins.

5.5.2.1 Trap Descriptions

SDC's generated traps are described in *Table 55* as they are displayed in the Web UI.

Table 55: SDC Generated Traps

Trap	Description
CpfChannelBindFailed	Indicates that a virtual server or SCTP client could not bind to an IP address and port, thus preventing client connection
CpfChannelBindFailedClear	Indicates that a virtual server or SCTP client that previously could not bind to an IP address and port, can now bind to an IP address and port
CpfConcurrentSessionsOverload	Indicates that the concurrent session utilization has reached its full capacity
CpfConcurrentSessionsOverloadClear	Indicates that the concurrent session utilization has dropped below the maximum
CpfDnsResolvingFailure	Indicates a DNS resolving (routing) failure
CpfDnsResolvingSuccess	Indicates a DNS resolving (routing) success
CpfLicenseAboutToExpire	Indicates the end of the license period
CpfLicenseAboutToExpireClear	Indicates the license period is valid
CpfLicenseClientRejected	Indicates that SDC is not licensed to accept clients
CpfMaxTracePerDayReached	Indicates that the number of daily traced transaction has reached the maximum threshold

CpfMaxTraceTPSReached	Indicates that the TPS rate has reached the maximum threshold
CpfMemorySizeOverload	Indicates that the memory utilization has reached its full capacity
CpfMemorySizeOverloadClear	Indicates that the memory utilization has dropped below the maximum
CpfNmsCollectingStatisticsFailure	Indicates that statistics collection has failed
CpfNmsCollectingStatisticsFailureClear	Indicates that statistics collection succeeded.
CpfNmsResourcesAlarm	Indicates the current usage status of one of the system resources
CpfNodeClientsQueueHighWatermark	Indicates that the size of MessageExecutor incoming queue or AsyncTaskExecutor queue exceeded the predefined high watermark (by default: 50% of the queue size)
CpfNodeClientsQueueLowWatermark	Indicates that the size of MessageExecutor incoming queue or AsyncTaskExecutor queue dropped below the predefined low watermark (by default: 10% of the queue size)
CpfNodeClientsIncomingQueueOverload	Indicates that that a Client Remote Peer’s incoming queue utilization has exceeded the maximum threshold
CpfNodeClientsIncomingQueueOverloadClear	Indicates that that a Client Remote Peer’s incoming queue utilization has dropped below the maximum
CpfNodeStateChangedShutDown	Indicates that the SDC Component’s state has changed to “Shutdown”
CpfNodeStateChangedWakeUp	Indicates that the SDC Component’s state has changed to “Wake Up”

cpfPeerRateLimitState	Indicates the threshold for TPS (transactions per second) rate limit for a peer
CpfPeerStateChangedChannelDown	Indicates that the channel between SDC and the Server Remote Peer’s is down
CpfPeerStateChangedChannelUp	Indicates that the channel between SDC and the Server Remote Peer’s is up
CpfPeerStateChangedServiceDown	Indicates that the Server Remote Peer’s Service is down
CpfPeerStateChangedServicePatialDown	Indicates that an active server peer is now partially out of service
CpfPeerStateChangedServiceUp	Indicates that the Server Remote Peer’s Service is up
CpfPoolHealthStateChangedGreen	Indicates that the pool state, which is based on the average of its peers’ health, is in the 80-100% range
CpfPoolHealthStateChangedRed	Indicates that the pool state, which is based on the average of its peers’ health, is in the 0-20% range
CpfPoolHealthStateChangedYellow	Indicates that the pool state, which is based on the average of its peers’ health, is in the 20-80% range
cpfPoolRateLimitState	Indicates the threshold for TPS (transactions per second) rate limits for a pool
CpfProxyGroupActiveProxyChanged	Indicates that the active proxy in a proxy group has changed
CpfRoutingFailed	Indicates that a message failed to reach its destination
CpfScriptInvocationFailed	Indicates that a script invocation has failed, and specifies the reason for the failure

CpfSctpLinkDown	Indicates a Multi-Homed SCTP Link is down
CpfSctpLinkDownUp	Indicates a Multi-Homed SCTP Link is up
CpfSiteConnectivityDown	Indicates that the connection to the EMS site is down.
CpfSiteConnectivityUp	Indicates that a connection to the EMS site that was previously down is now up
CpfSiteReplicationTargetDown	Indicates that a Remote Server used for site replication is down
CpfSiteReplicationTargetUp	Indicates that a Remote Server uses for site replication is up
CpfTripoIsDown	Indicates that the connection between CPF and Tripo was disconnected
CpfTripoIsDownClear	Indicates that the connection between CPF and Tripo was restored
CpfUserAuthenticationFailure	Indicates that a user login attempt has failed.
SlfAgentAddFailure	Note: This trap is not currently available.
SlfAgentOutOfMemory	Note: This trap is not currently available.
SlfAgentStartedSuccesssfully	Note: This trap is not currently available.
Ss7LicenseAlarmActive	Indicates that the SS7 driver is processing more TPS than defined in the SS7 license.
Ss7LicenseAlarmInactive	Indicates that the SS7 driver is processing an amount of TPS that is within the SS7 license definition.
UploaderAgentFinishRequest	Indicates that there are no degraded files to upload
UploaderAgentSftpConnFailure	Indicates that degraded files cannot be uploaded due to SFTP connection failure

UploaderAgentSftpUploadFailure	Indicates that degraded files cannot be uploaded
UploaderReceiverRequest	Indicates that the system received a SOAP request to start uploading degraded files to a predefined destination

5.5.2.2 Configuring the Dilution Manager

This section describes how to configure the dilution manager parameters for a specific alarm, so that it can be enabled.

To configure the SNMP alarm dilution parameter values for a specific alarm:

1. Go to > **Administration** > **SNMP** > **SNMP Dilution Manager**.

   The SNMP Dilution Manager table displays a list of SNMP alarms and their dilution parameters.

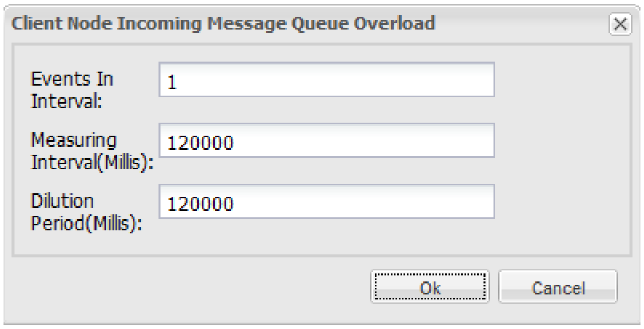
   Table 57: Trap Viewer Table*Table 57* describes the SNMP Dilution Manager Table parameters.

Table 56: SNMP Dilution Manager Table

Column	Description
Alarm 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). The value “0” disables the trap.
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)

2. Select the alarm that you want to edit.
3. Click **Edit**. The alarm properties window appears for the selected alarm.

Figure 63: Alarm Properties



Events In Interval: 1

Measuring Interval(Millis): 120000

Dilution Period(Millis): 120000

4. Edit the relevant parameter values.

5. Click **OK**.

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**

5.5.4 Monitoring SNMP Traps

The **Monitoring** tab allows you to monitor the real-time status of services and resources in your system.

5.5.4.1 Trap Viewer

The trap viewer provides real-time monitoring of SNMP traps generated by SDC.

To view an SNMP trap:

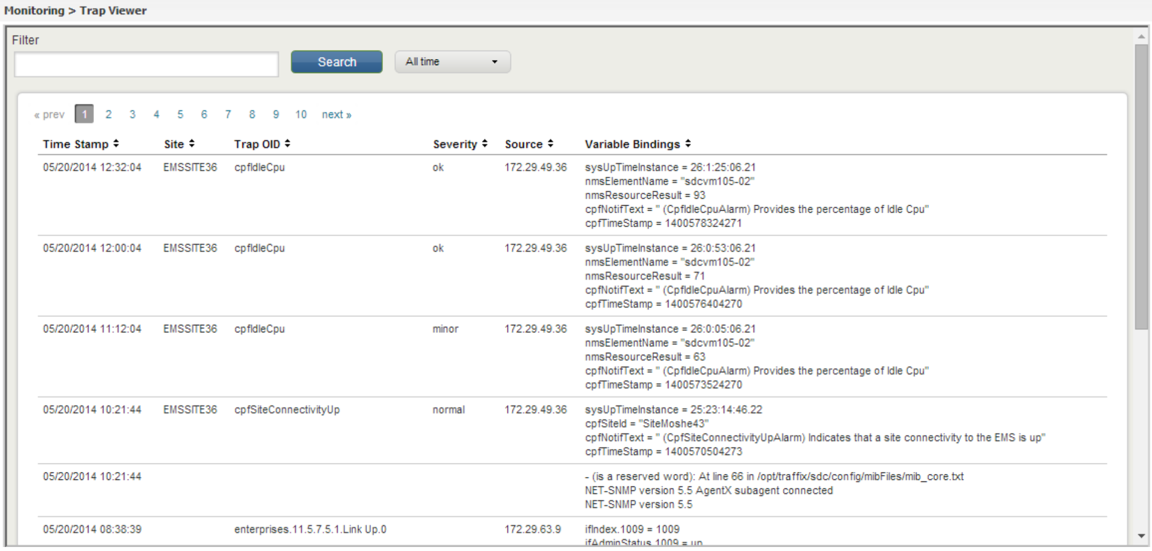
1. Go to **Monitoring** > **Trap Viewer**. The Trap Viewer screen is displayed, as shown in *Figure 64*.

2. To display specific information:

   a. Using the Filter text box, enter the value that you wish to filter the displayed traps by, and click **Search**.

b. Using the drop-down list next to the **Search** button, select the time period that you wish to display traps for.

Figure 64: Trap Viewer



Time Stamp	Site	Trap OID	Severity	Source	Variable Bindings
05/20/2014 12:32:04	EMSSITE36	cpfIdleCpu	ok	172.29.49.36	sysUpTimeInstance = 26:1:25:06.21 nmsElementName = "sdcvm105-02" nmsResourceResult = 93 cpfNotifText = " (CpfIdleCpuAlarm) Provides the percentage of Idle Cpu" cpfTimeStamp = 1400578324271
05/20/2014 12:00:04	EMSSITE36	cpfIdleCpu	ok	172.29.49.36	sysUpTimeInstance = 26:0:53:06.21 nmsElementName = "sdcvm105-02" nmsResourceResult = 71 cpfNotifText = " (CpfIdleCpuAlarm) Provides the percentage of Idle Cpu" cpfTimeStamp = 1400576404270
05/20/2014 11:12:04	EMSSITE36	cpfIdleCpu	minor	172.29.49.36	sysUpTimeInstance = 26:0:05:06.21 nmsElementName = "sdcvm105-02" nmsResourceResult = 63 cpfNotifText = " (CpfIdleCpuAlarm) Provides the percentage of Idle Cpu" cpfTimeStamp = 1400573524270
05/20/2014 10:21:44	EMSSITE36	cpfSiteConnectivityUp	normal	172.29.49.36	sysUpTimeInstance = 25:23:14:46.22 cpfSiteId = "SiteMoshe43" cpfNotifText = " (CpfSiteConnectivityUpAlarm) Indicates that a site connectivity to the EMS is up" cpfTimeStamp = 1400570504273
05/20/2014 10:21:44					- (is a reserved word): At line 66 in /opt/traffix/sdc/config/mibFiles/mib_core.txt NET-SNMP version 5.5 AgentX subagent connected NET-SNMP version 5.5
05/20/2014 08:38:39		enterprises.11.5.7.5.1.Link Up.0		172.29.63.9	ifIndex.1009 = 1009 ifAdminStatus.1009 = up

Table 57 shows the trap viewer table columns:

Table 57: Trap Viewer Table

Column	Description
Time Stamp	The time in which the trap was generated.
Site Id	The site that the trap was generated on.
Trap OID	The ID/name of the trap in the SNMP MIB file.
Source	The origin (IP address) of the SNMP trap.
Variable Bindings	The parameters of the trap, as detailed in the *F5 SDC SNMP Guide.*

5.6 System View

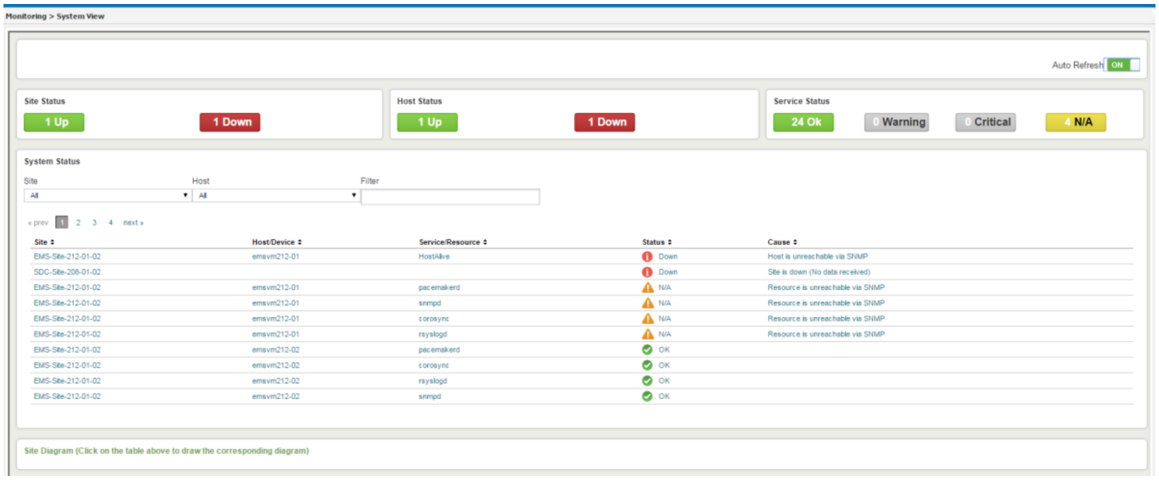
The System View provides a real-time global view of the system resources:

To view the system resources:

1. Go to **Monitoring** > **System View**. The System View screen is displayed.

Note: To disable auto refresh of the screen data, switch the **Auto Refresh** button from ON to OFF. Auto-refresh is enabled by default, and must be disabled every time the screen is accessed.

Figure 65: System View



Site	Host/Device	Service/Resource	Status	Cause
EMS-Site-212-01-02	emsvm212-01	HostAlive	Down	Host is unreachable via SNMP
SDC-Site-208-01-02			Down	Site is down (No data received)
EMS-Site-212-01-02	emsvm212-01	pacemakerd	N/A	Resource is unreachable via SNMP
EMS-Site-212-01-02	emsvm212-01	snmpd	N/A	Resource is unreachable via SNMP
EMS-Site-212-01-02	emsvm212-01	corosync	N/A	Resource is unreachable via SNMP
EMS-Site-212-01-02	emsvm212-01	rsyslogd	N/A	Resource is unreachable via SNMP
EMS-Site-212-01-02	emsvm212-02	pacemakerd	OK	
EMS-Site-212-01-02	emsvm212-02	corosync	OK	
EMS-Site-212-01-02	emsvm212-02	rsyslogd	OK	
EMS-Site-212-01-02	emsvm212-02	snmpd	OK	

Table 58 provides a legend of the different monitoring screen panes.

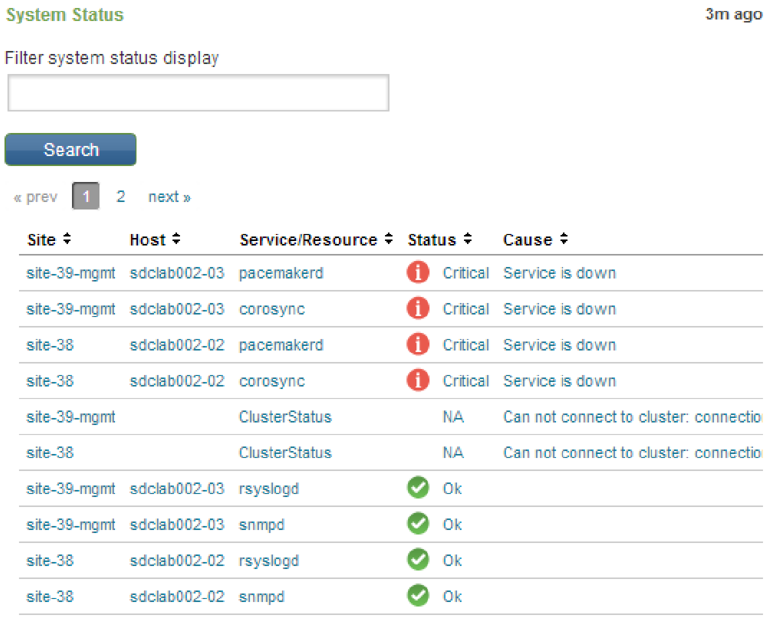
Table 58: System View

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

Pane	Description
	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 (as shown in *Figure 65*)
Site Diagram	Displays the selected site’s diagram, detailing the hosts, services and resources (selected in the system status table).

Figure 66 shows a more detailed view of the System Status table.

Figure 66: System Status Table



System Status

3m ago

Filter system status display

Site ⬍	Host ⬍	Service/Resource ⬍	Status ⬍	Cause ⬍
site-39-mgmt	sdclab002-03	pacemakerd	ⓘ Critical	Service is down
site-39-mgmt	sdclab002-03	corosync	ⓘ Critical	Service is down
site-38	sdclab002-02	pacemakerd	ⓘ Critical	Service is down
site-38	sdclab002-02	corosync	ⓘ Critical	Service is down
site-39-mgmt		ClusterStatus	NA	Can not connect to cluster: connectio
site-38		ClusterStatus	NA	Can not connect to cluster: connectio
site-39-mgmt	sdclab002-03	rsyslogd	✓ Ok	
site-39-mgmt	sdclab002-03	snmpd	✓ Ok	
site-38	sdclab002-02	rsyslogd	✓ Ok	
site-38	sdclab002-02	snmpd	✓ Ok	

Table 59 provides a legend of the system status table:

Table 59: System Status Table

Column	Description
Site	The name of the site to which the service/resource belongs
Host/Device	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
Cause	The cause of the service/resource status

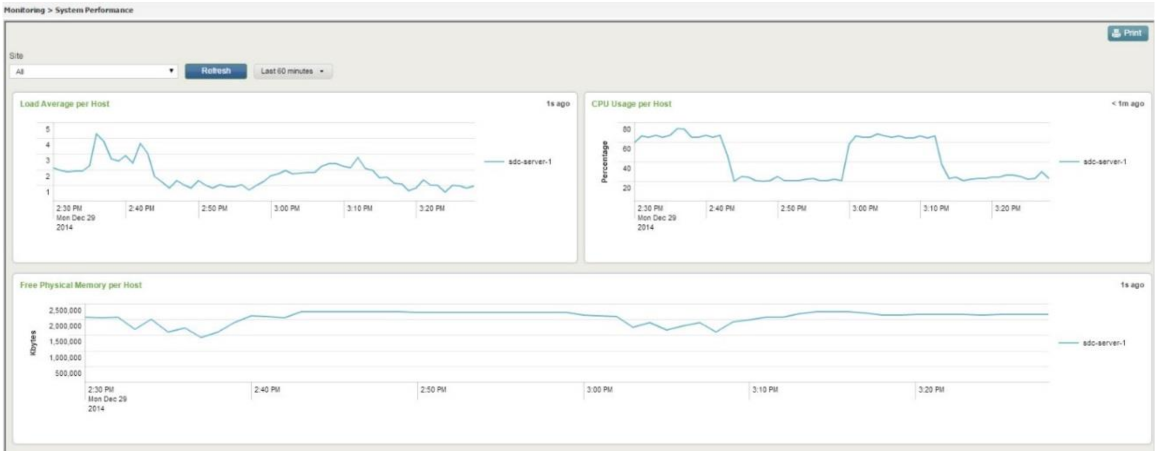
5.7 System Performance

The System Performance provides a real-time view of the system performance.

To view system performance:

1. Go to **Monitoring** > **System Performance**. The System Performance screen is displayed, as shown in *Figure 67*.
2. To display specific information:
   a. Using the **Site** drop-down list, select a specific site to display data for.
   b. Using the drop-down list next to the **Refresh** button, select the time period that you want to display data for.

Figure 67: System Performance



Monitoring > System Performance

Site: All

Last 60 minutes

Table 60 provides a legend of the available system performance graphs:

Table 60: System Performance Graphs

Column	Description
Load average per host	Depicts the average load (an integer value, representing the load average of the host)
CPU usage per host	Depicts the CPU usage (used percentage) per host
Free physical memory per host	Depicts the free memory (in kilobytes) per host

5.7.1 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 Tripo errors, respectively, is under:

/opt/traffix/sdc/logs/cpf1/session_output

/opt/traffix/sdc/logs/cpf1/session_error

When enabled, the following events/errors are written to the log files with the information shown in *Table 61*.

Table 61: 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 (Master /Slave) Note*:* A persistent session that has no binding key will appear as a master session. ▪ Master Session ID

Event	Related information written to the log file
	Note: This is displayed for slave sessions only. ▪ SM Row ID ▪ Binding Keys Note: This is displayed for master sessions only. ▪ 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

Event	Related information written to the log file
	▪ 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 ▪ 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 ▪ BF – binding failure ▪ 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**.**

To add session attributes to a session log:

1. 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 to flush the remaining log data to the log file.

5.8 Host Performance

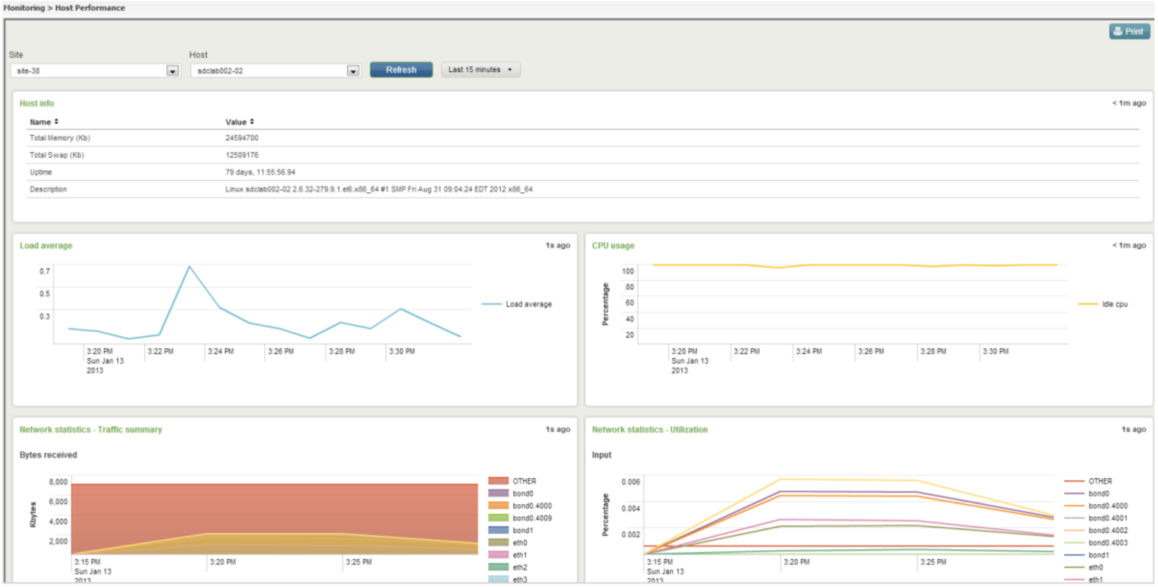
The Host Performance provides real-time view of a host’s performance.

To view host performance:

1. Go to **Monitoring** > **Host Performance**. The Host Performance screen is displayed, as shown in *Figure 68.*

2. Using the drop-down list next to the **Refresh** button, select the time period that you want to display data for.

Figure 68: Host Performance



Monitoring > Host Performance

Site: site-38

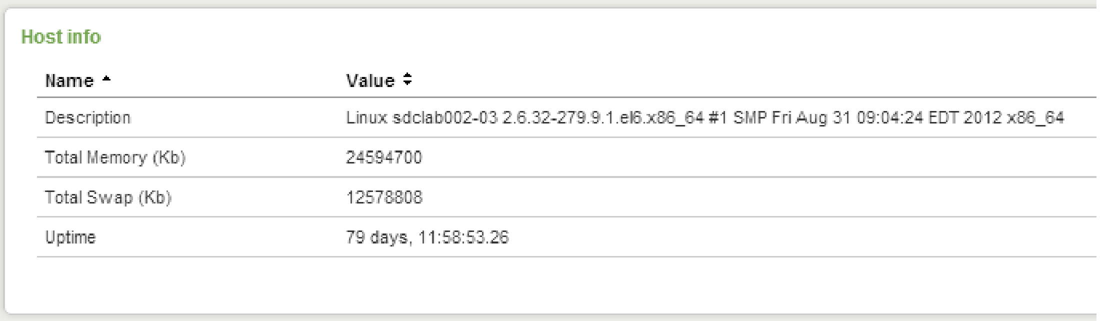
Host: sdclab002-02

Host info — < 1m ago

Name	Value
Total Memory (Kb)	24594700
Total Swap (Kb)	12509176
Uptime	79 days, 11:55:56.94
Description	Linux sdclab002-02 2.6.32-279.9.1.el6.x86_64 #1 SMP Fri Aug 31 09:04:24 EDT 2012 x86_64

3. From the upper part of the screen, select the **Site** and **Host**. The display is refreshed and the Host Info pane changes according to the selected site and host, as shown in *Figure 69*.

Figure 69: Host Info



Host info

Name ▲	Value ⬍
Description	Linux sdclab002-03 2.6.32-279.9.1.el6.x86_64 #1 SMP Fri Aug 31 09:04:24 EDT 2012 x86_64
Total Memory (Kb)	24594700
Total Swap (Kb)	12578808
Uptime	79 days, 11:58:53.26

4. Using the drop-down list next to the **Refresh** button, select the time period that you want to display data for.

Table 62 provides a legend of the available system performance graphs:

Table 62: Host Performance Graphs

Column	Description
Load average	The average system load for the host.
CPU usage	The percentage of CPU that was used by the host.
Network statistics – Traffic Summary	Depicts the traffic (in bytes) received and sent by the host.
Network statistics – Utilization	Depicts the percentage of network utilization used by the host’s output and input.
Disk I/O	Depicts the hosts’ read and written bytes
Disk space usage by partition	Depicts the disk space usage of the host’s partitions.
Memory usage	Depicts the free/cached/buffered/Available swap memory (in kilobytes) of the host

5.9 System History Status

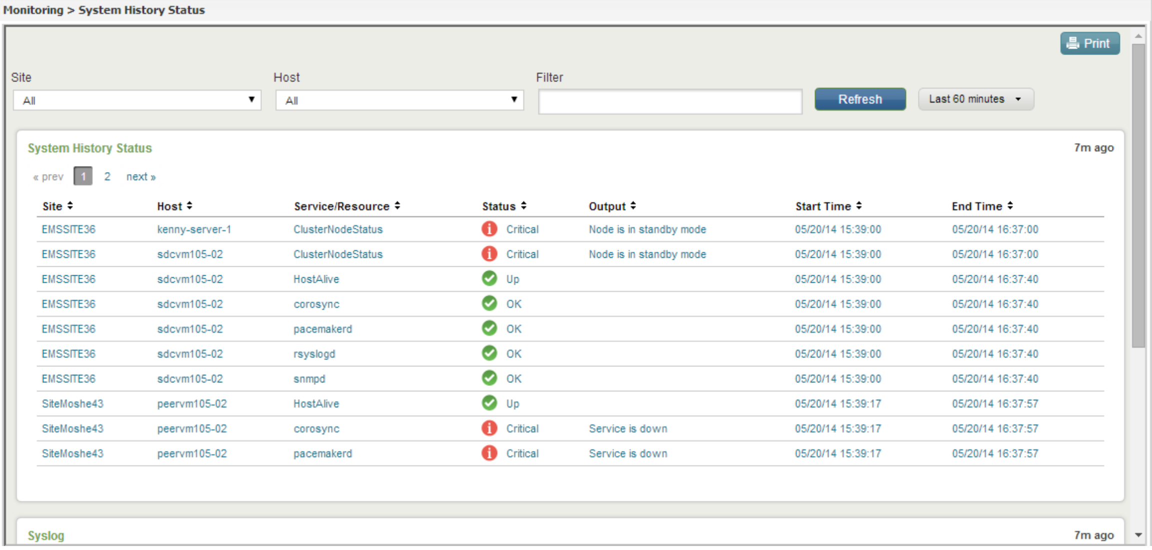
The System History Status provides a real-time global view of the system resources:

To view the system resources:

1. Go to **Monitoring** > **System History Status**. The System History Status screen is displayed, as shown in *Figure 70*.
2. To display specific information in the System History Status table:
   a. Using the **Site** and/or **Host** drop-down lists, select a specific site and/or host to display data for.
   b. Using the **Filter** text box, enter the value that you want to filter the displayed information by.
   c. Using the drop-down list next to the **Refresh** button, select the time period that you wish to display data for.

The log messages produced in the selected time resolution will be displayed in the Syslog pane.

Figure 70: 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 63 provides a legend of the System History Status table:

Table 63: 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
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.

5.10 Logging and Syslog

The SDC events are logged according to their nature (e.g.: system, networking, etc.). Log messages (FEP and CPF) 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.

5.10.1 Setting the Log Levels

You can set the log level depending on the detail level that you want to log.

To set the SDC log levels:

1. Go to **Administration** > **Specific Site Settings** > **Logging**.
2. From the **Log Level** drop-down list, select the log level for all log messages, by selecting it from the drop-down list.

Note: SDC prints all logs of the selected log level and also those of above log levels.

Table 64 describes the different Log Detail Levels.

Table 64: 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

Level	Description
	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.
Trace	Indicates events that are most useful to debug an application with, at a finer-grained information level than the Debug level.

3. Alternatively, select the **Customize Log Level** checkbox and then 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.

Table 65: 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

Log Category	Description
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)**

Note: The log level of a category cannot extend the log level as defined in log4j.xml.

4. Click **Submit** to save the log settings.

5.10.2 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 CPF and FEP log messages can be configured to be sent to a remote Syslog Daemon.

To add an SDC Syslog Addresses:

1. Go to **Administration** > **Specific Site Settings** > **Logging** > **Syslog Addresses**.

   The Syslog Addresses table appears.

   Table 66 presents a list of Syslog Daemon Addresses properties.

Table 66: Syslog Addresses

Column	Description
IP Address	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.**
3. Repeat this step for any additional Syslog Daemons that should receive the log message output.
4. Click **Submit** to save the log settings.

5.10.3 Log File Size Control

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 67* shows their default values.

Table 67: Log File Size

Parameter	Default Value
MaxBackupIndex	10
MaxFileSize	10MB

5.11 🌐 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*).

5.11.1 Configuring a Tracing Rule

Before configuring a tracing rule to capture transaction data, you need to define the relevant tracing attributes.

5.11.2 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 tab 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 **FilterType**, select the data type of the new attribute. e.g.: String
6. Under **Description**, type in a short description of the attribute.
7. Click **Submit**.

Note: For additional information on the decision table attributes, see *Appendix D: Decision Table Attributes*.

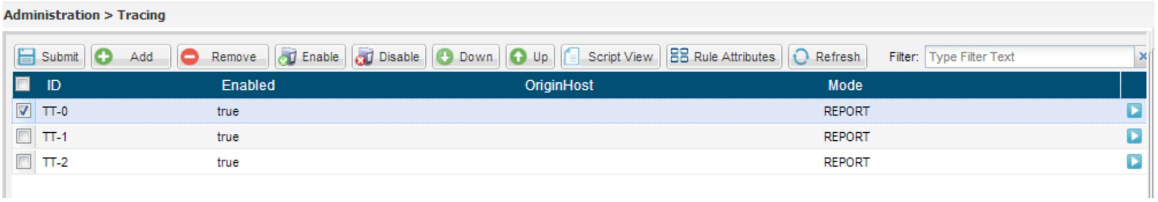
5.11.3 Adding a Tracing Rule

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 71: Tracing Rules



□	ID	Enabled	OriginHost	Mode
☑	TT-0	true		REPORT
□	TT-1	true		REPORT
□	TT-2	true		REPORT

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**
   - **REPORT AND LOG**
   - **REPORT AND LOGWITH 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.

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

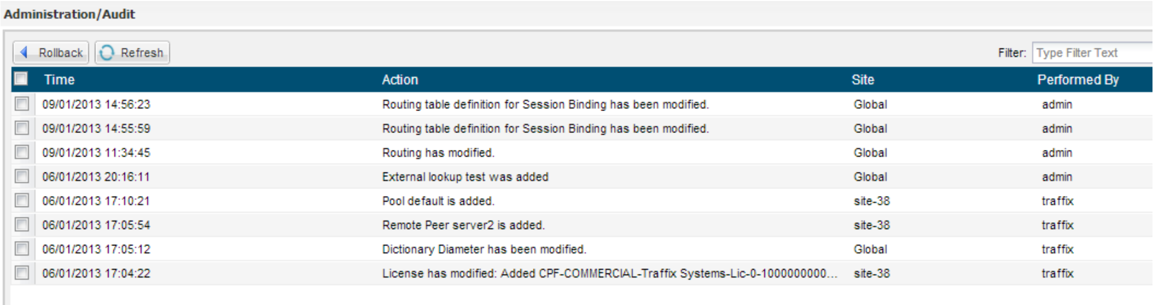
- Changing a Flow script
- Changing a Health Monitor
- Changing a Routing Script
- Change a Transformation Script
- Change a User Tracing script
- Changing a Cluster Node’s configuration
- Editing a Pool
- Editing a Remote Node
- Removing a Remote Node
- Removing a Pool
- Removing a User
- Deleting a 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

To view the audited entries:

1. Go to **Administration** > **Audit.** The Audit screen is displayed.

Note: The maximum number of audit entries that are displayed is 200.

Figure 72: Audit



☐	Time	Action	Site	Performed By
☐	09/01/2013 14:56:23	Routing table definition for Session Binding has been modified.	Global	admin
☐	09/01/2013 14:55:59	Routing table definition for Session Binding has been modified.	Global	admin
☐	09/01/2013 11:34:45	Routing has modified.	Global	admin
☐	06/01/2013 20:16:11	External lookup test was added	Global	admin
☐	06/01/2013 17:10:21	Pool default is added.	site-38	traffix
☐	06/01/2013 17:05:54	Remote Peer server2 is added.	site-38	traffix
☐	06/01/2013 17:05:12	Dictionary Diameter has been modified.	Global	traffix
☐	06/01/2013 17:04:22	License has modified: Added CPF-COMMERCIAL-Traffix Systems-Lic-0-1000000000...	site-38	traffix

Table 68 presents a list of audited actions taken by SDC users.

Table 68: 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.

To refresh the Audit table:

1. Click **Refresh**.

To restore a previous configuration mode:

1. Select the Audit entry you want to rollback.
2. Click **Rollback**.

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.

🔔 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).

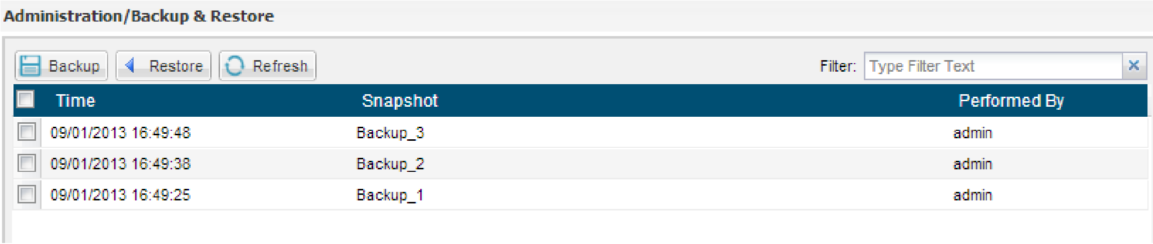
6.1.2 Backup & Restore

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 & Restore**. The Backup & Restore screen is displayed:

Figure 73: Backup & Restore



☐	Time	Snapshot	Performed By
☐	09/01/2013 16:49:48	Backup_3	admin
☐	09/01/2013 16:49:38	Backup_2	admin
☐	09/01/2013 16:49:25	Backup_1	admin

Table 69 presents a list of backup snapshots actions taken by the SDC users.

Table 69: 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.

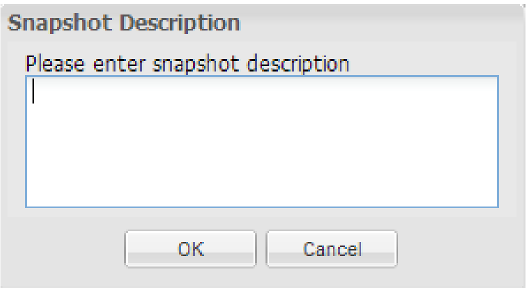
To refresh the Backup & Restore table:

1. Click **Refresh**.

To backup the current configuration and create a snapshot of SDC:

1. Click **Backup**. The Snapshot Description dialog box is displayed.

Figure 74: Snapshot Description



Snapshot Description

Please enter snapshot description

2. Enter a meaningful description for the current SDC configuration.
3. Click **OK**. The new backup snapshot appearss in the Backup Snapshots table.

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 with 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, authorizes actions of users according to their given roles, and supports addition of new users, and removal and editing of existing ones.

Table 70 details the user roles and their privileges:

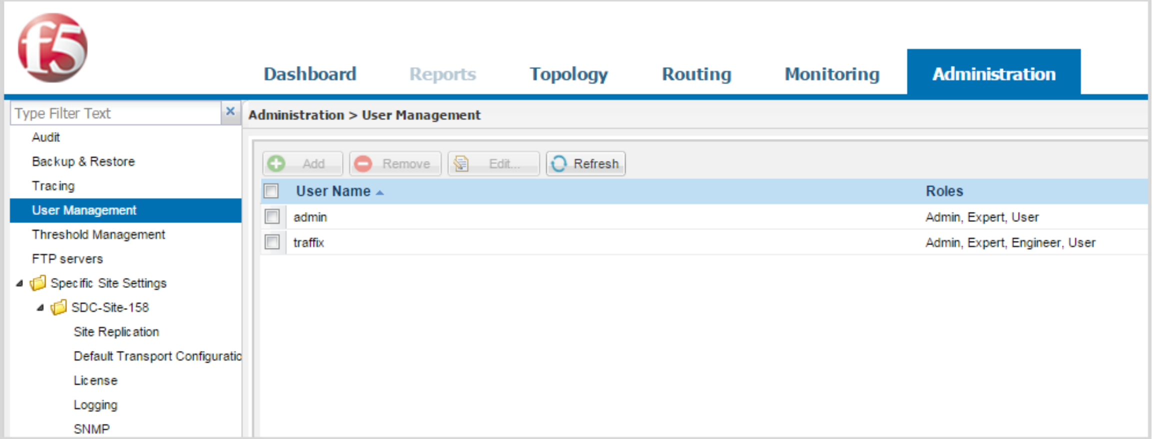
Table 70: User Type Privileges

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.

To create a new user in SDC Web UI:

1. Go to **Administration** > **User Management**. The User Management screen is displayed.

Figure 75: User Management



Administration > User Management

□	User Name	Roles
□	admin	Admin, Expert, User
□	traffix	Admin, Expert, Engineer, User

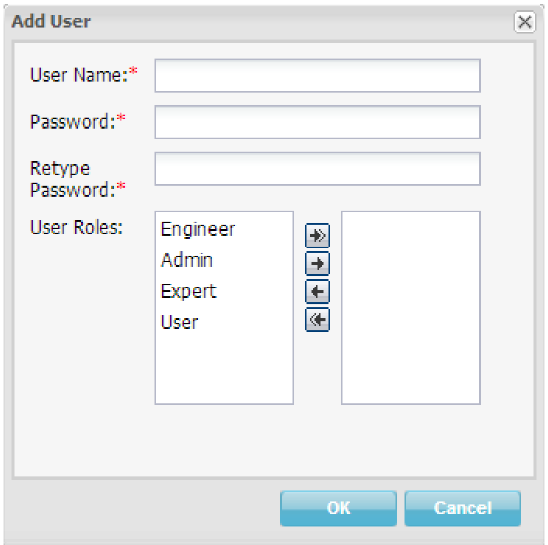
Table 71 presents a list of the SDC users.

Table 71: SDC Users

Column	Description
User Name	The user’s unique identifier.
Roles	The list of privileges the user is assigned with.

2. Click **Add**. The Add User dialog box appears

Figure 76: Add User



Add User

User Name:*

Password:*

Retype Password:*

User Roles:

Engineer
Admin
Expert
User

3. In **User Name** field, enter the user’s unique identifier.

4. In **Password** field, enter the user’s password and retype it in Retype Password field.

5. From the **User Roles** box, click to select the role you want to add to the user’s role list.

6. Click the **single right arrow** button. The role is added to user’s role list.

7. 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.

8. To add all available roles to user’s role list, click the **double right arrow** button.

9. To remove a role from the role list, click to select it from the right box and then click the **left arrow** button. To remove all roles from the role list click the **double left arrow** button.

10. Click **OK**.

To remove any user from the list:

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

To refresh the user list:

1. Click **Refresh**.

To edit a user:

1. Select the user from the **User Name** list and click **Edit**. The Edit User dialog box appears.

Note: You may edit all fields, as detailed in the above steps.

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*.

Appendix A: User Data Storage

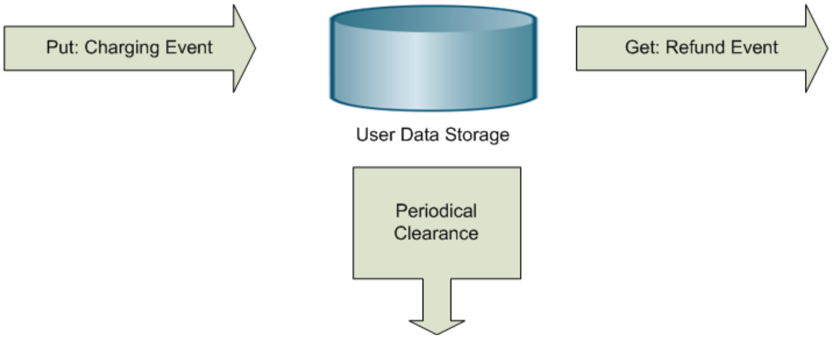
SDC allocates a special memory hook on which you may create and maintain simple and complex data structures. The memory hook is called User Data Storage. The User Data Storage is typically used to store cross-session data (e.g. client details).

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 77: 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:
   - public static StorageProviderFactory getInstance();
2. Creating a table:
   - public <K, V> StorageProvider<K, V> createUserTable(String tableName);
   - public <K, V> StorageProvider<K, V> createUserTable(String tableName, long lifespan);
3. Performing table operations (see the following implementation example)
4. 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 72: 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.	Specified 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 73 describes the Supported Application Identifiers.

Table 73: 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

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:

1. Configure a file server by performing the following steps:
2. Go to **Topology** > Remote Peers.
3. Click **Add**. The Add Peer wizard appears.
4. In the **Name** field, set the name for this peer.
5. In the **Protoco**l field, select File.
6. Click **Next**.
7. Set **Primary IP** of the File Server.
8. Set **Primary Port** of the File Server.
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. Click **Finish**.
13. 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. For more information about configuring routing rules, see *Configuring a Routing Rule*.

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.

The "Null" property can appear in the event. This support for the "null" value checks if the attribute exists in a message or not, and can be used for either string or octet string AVPs.

Checking if an AVP exists is performed by typing "null" in the value field.

Table 74: Decision Table 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

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
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 balancing policy	session.setContextId()
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	

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
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()	
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	session.isSticky()	Set stickiness mode on session	session.setIsSticky()

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
			False, the session's routing should be calculated per message			
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 with Accounting-Record-Type STOP/EVENT)	session.shouldAutomaticallyRelease()	Sets automatic release of the session	session.setShouldAutomaticallyRelease()
Session	IDLE_SESSION_TIMEOUT	Boolean	Should update session timeout upon request arrival	session.shouldRefreshTimeoutOnGet()	Sets refreshing policy	session.setShouldRefreshTimeoutOnGet()

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
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	
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:	peer.getProtocol()	Cannot set	

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
			Protocol.Diameter)			
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 domain or its profile	peer.getMetaData().getLocalConfiguredRealm()	Cannot set	

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
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	

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
Diameter message	IMSI	String	The Subscription-Id-Data when type is END_USER_IMSI	message.getImsi()	Cannot set	
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	

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
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	
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:	stack.getState()	cannot set	cannot set	

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
		State.OPEN, State.CLOSE				
Stack	UID	Instance ID of node		cannot set	Cannot set	
Stack	CPF_GROUP_NAME	Group name of node		cannot set	Cannot set	
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

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
HTTP Answer	REASON_PHRASE	String	The HTTP Answer's reason description	message.getProperty("REASON_PHRASE")	Cannot set	Cannot set
SS7 Message	OPERATION_CODE	Integer	TCAP Component (usually GSM-MAP) command code	message.getProperty("OPERATION_CODE")	N/A	Cannot set

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
SS7 Message	ERROR_CODE	Integer	TCAP Component (usually GSM-MAP) error code	message.getProperty("ERROR_CODE")	N/A	Cannot set
SS7 Message	DESTINATION_ROUTE_ON_GT	Boolean	Shall message be routed by SCCP layer according to the Global Title	message.getProperty("DESTINATION_ROUTE_ON_GT")	N/A	message.setProperty("DESTINATION_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_NUMBER	Integer	The numbering plan attribute of the	message.getProperty("DESTINATION_GT_NUMB	N/A	message.setProperty("DESTINATION_GT_N

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
	ING_PLAN		destination global title	ERING_PLAN")		UMBERING_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")
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	Integer	The GT Indicator attribute of the	message.getProperty("ORIGINATION_R	N/A	message.setProperty("ORIGINATION_ROUT

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
	_ON_GT		destination global title	OUTE_ON_GT")		E_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")
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_S	Integer	The encoding scheme attribute of the origin global title	message.getProperty("ORIGINATION_GT_ENCODING_SCHEME")	N/A	message.setProperty("ORIGINATION_GT_ENCODING_SCHEME")

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
	CHEME					
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")
SS7 Message	DESTINATION_SSN	Integer	The Destination Subsystem Number	message.getProperty("DESTINATION_SSN")	N/A	message.setProperty("DESTINATION_SSN")

Element	Property	Returned Value Type	Reading	Groovy equivalent (Read)	Writing	Groovy equivalent (Write)
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

Note: The SDC support LDAPv3 and lower.

To enable user login using an external LDAP server:

1. Edit the following attributes (in the table below) in the *ldap-config.properties* file (as applicable):

Note: 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 75: 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

Attribute	Description	Mandatory	Example
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
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})

Attribute	Description	Mandatory	Example
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
authenticationStrategy	The authentication processing behavior.	Yes	default – defines clear text and SSL startTLS – defines the start TLS behavior SSL – defines SSL with a certificate
trust.store	The location of the security certificate.	Yes Note: Only for the startTLS and SSL authentication strategies	C\:\\Temp\\sslkey.jks

Attribute	Description	Mandatory	Example
trust.store.password	The password of the security certificate.	Yes Note: Only for the startTLS and SSL authentication strategies	ENC(W4WStHUig4GJkm5QR2PNacoFQb8Fcbu1)

2. Update the security file with the LDAP security file by performing the following steps:

   a. Go to */opt/traffix/sdc/config/security/LDAP/* and copy the applicationContext-security.xml security file.

   b. Go to */opt/traffix/sdc/utils/apache-tomcat/webapps/MgmtConsole/WEB-INF/and paste the applicationContext-security.xml* security file.

3. Create a CA certificate and Server certificate by performing the following steps on the LDAP server:

   a. Run the following commands:

```
cd /etc/openldap/certs
mkdir new
cd ./new
certutil -N -d .
```

   b. Generate a CA certificate by running the following commands:

```
certutil -S -n "ldap-ca.lab.traffixsystems.local" \
-s cn=ldap-ca.lab.traffixsystems.local \
-2 -x -t "CT,," -m 1000 -v 120 -d . -k rsa
```

   c. Generate a Server certificate by running the following commands:

```
certutil -S -n "ldap-server.lab.traffixsystems.local" \
-s cn=ldap-server.lab.traffixsystems.local \
-c "ldap-ca.lab.traffixsystems.local" \
-t "u,u,u" -m 1001 -v 120 -d . -k rsa

certutil -L -d . -a -n ldap-server.lab.traffixsystems.local > ldap-server.pem
certutil -L -d . -r -n ldap-server.lab.traffixsystems.local > ldap-server.der
certutil -L -d . -a -n ldap-ca.lab.traffixsystems.local > cacert.pem
certutil -L -d . -r -n ldap-ca.lab.traffixsystems.local > cacert.der

pk12util -d . -o ldap-server.p12 -n ldap-server.lab.traffixsystems.local
```

d. Add the following parameters to the */etc/openldap/slapd.conf* file:

- TLSCipherSuite HIGH:MEDIUM:+TLSv1:!SSLv2:+SSLv3
- TLSCACertificatePath /etc/openldap/certs
- TLSCertificateFile ldap-server.lab.traffixsystems.com
- TLSVerifyClient never

e. Run the following command to translate the configuration from the *slapd.conf* to the *slapd.d*.folder:

```
slaptest -f /etc/openldap/slapd.conf -F /etc/openldap/slapd.d
```

f. Add the following lines to the */etc/openldap.conf* file after the modulepath directive:

```
TLS_CACERT /etc/openldap/certs/cacert.pem
TLS_REQCERT allow
```

4. Restart the OpenLDAP server by running the following command:

service slapd restart

5. Import the CA certificate into the Java keystore by performing the following steps on the SDC server:

a. Run the following script:

```
!/bin/bash
```

```
JAVA_HOME=/usr/jdk/latest
KEYTOOL=$JAVA_HOME/bin/keytool
STOREPASS="traffix"
CA_PUBLIC_CERT=./cacert.der
keystore_file=./keystore.jks

echo "# Importing DemoCA Public Certificate ${CA_PUBLIC_CERT} as trusted"
$KEYTOOL \
-import \
-trustcacerts \
-alias "ldap-ca.lab.traffixsystems.local" \
-file ${CA_PUBLIC_CERT} \
-storepass $STOREPASS \
-keystore $keystore_file

exit 0
```

b. Copy the modified *keystore.jks* file to the */opt/traffix/sdc/keystore* folder on each SDC server.

c. Add the following values to the **JAVA_CONFIG_MGR_OPTS** parameter in the */opt/traffix/sdc/bin/traffix_webui_init* file:

- -Djavax.net.ssl.trustStore=<path to truststore file>
- -Djavax.net.ssl.trustStorePassword=<password for truststore>

6. Start the SDC.

Glossary

The following table lists the terms and abbreviations used in this document.

Table 76: Terms and Abbreviations

Term	Definition
AAA	Authentication, Authorization and Accounting
ACL	Access Control List
AF	Application Function
Answer	A message sent from one Client/Server Peer to the other following a request message
API	Application Programming Interface
AVP	Attribute Value Pair
CLI	Command Line Interface
Client Peer	A physical or virtual addressable entity which consumes AAA services
CPF	Control Plane Function
Data Dictionary	Defines the format of a protocol’s message and its validation parameters: structure, number of fields, data format, etc.
DEA	Diameter Edge Agent
Destination Peer	The Client/Server peer to which the message is sent
DRA	Diameter Routing Agent
EMS Site	Element Management System Site
FEP-In	In-Front End Proxy
FEP-Out	Out-Front End Proxy

Term	Definition
Geo Redundancy	A mode of operation in which more than one geographical location is used in case one site fails
HA	High Availability
HSS	Home Subscriber Server
HTTP	Hypertext Transfer Protocol
IMS	IP Multimedia Subsystem
JMS	Java Message Service
KPI	Key Performance Indicator
LDAP	Lightweight Directory Access Protocol
LTE	Long Term Evolution
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)
MME	Mobility Management Entity
NGN	Next Generation Networking
Node	Physical or virtual addressable entity
OAM	Operation, Administration and Maintenance
OCS	Online Charging System
Origin Peer	The peer from which the message is received
PCEF	Policy and Charging Enforcement Function
PCRF	Policy and Charging Rules Function
PLMN	Public Land Mobile Network
Pool	A group of Server Peers
RADIUS	Remote Authentication Dial In User Service

Term	Definition
Request	A message sent from one Client/Server peer to the other, followed by an answer message
SCCP	Signaling Connection Control Part
SCTP	Stream Control Transmission Protocol
SDC	Signaling Delivery Controller
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
SNMP	Simple Network Management Protocol
SS7	Signaling System No. 7
TCP	Transmission Control Protocol
TLS	Transport Layer Security
Transaction	A request message followed by an answer message
Tripo	Session data repository
UDP	User Datagram Protocol
UE	User Equipment
URI	Universal Resource Identification.
Virtual Server	A binding point used by SDC to communicate with the Remote Peers (Clients and Servers)
VPLMN	Visited Public Land Mobile Network
Web UI	Web User Interface
WS	Web Service