Policy Management

OSSI XML Interface Definitions Guide

910-6400-001 Revision A
June 2012

Table of Contents

List of Tables

Chapter 1

About This Guide

Topics:

This guide describes the Configuration Management Platform (CMP) OSSI XML interface. This interface allows an operator or third party system to programmatically push configuration information to and retrieve operational statistics from the policy server deployment.

Introduction

This guide describes the Configuration Management Platform (CMP) OSSI XML interface. This interface allows an operator or third-party system to programmatically push configuration information to and retrieve operational statistics from the policy server deployment.

Conventions

The following conventions are used throughout this guide:

- **Bold text** in procedures indicates icons, buttons, links, or menu items that you click on.
- *Italic text* indicates variables.
- `Monospace text` indicates text displayed on screen.
- `Monospace bold text` indicates text that you enter exactly as shown.

How This Guide is Organized

The information in this guide is presented in the following order:

- *About This Guide* contains general information about this guide, the organization of this guide, and how to get technical assistance.
- *Schema Definitions and Request Specifications* describes the Schema Definitions and Request Specifications used within the OSSI XML interface.
- *Common Responses and Commands* describes the generic response format for many of the commands used and the commands that are common to the various OSSI interfaces.
- *Topology Interface* describes the interface that allows users to manage and query network elements, paths, and other topology-related objects within the system.
- *Subscriber Interface* describes the interface that allows users to manage and query subscriber elements within their system.
- *Operational Measurements Interface Overview* describes the interface used to retrieve operational counters from the system.
- *Operational Measurement Requests* describes the various Operational Measurements (OM) groups and the individual OM statistics that comprise them.
- *Identity Management (IDM)* describes the interface that allows the CMP to configure user names, passwords and roles using the OSSI interface.
- *Policy Table Interface* describes an interface that allows you to generalize multiple similar policies into a single policy, simplifying processes such as adding new policies, modifying existing policies, and checking consistency among policies.

Scope and Audience

This guide is intended for operators or third party systems who are responsible for pushing configuration information to and retrieving operational statistics from a policy server deployment.

Documentation Admonishments

Admonishments are icons and text throughout this manual that alert the reader to assure personal safety, to minimize possible service interruptions, and to warn of the potential for equipment damage.

Table 1: Admonishments

	**DANGER**: (This icon and text indicate the possibility of *personal injury*.)
	**WARNING**: (This icon and text indicate the possibility of *equipment damage*.)
	**CAUTION**: (This icon and text indicate the possibility of *service interruption*.)

Customer Care Center

The Tekelec Customer Care Center is your initial point of contact for all product support needs. A representative takes your call or email, creates a Customer Service Request (CSR) and directs your requests to the Tekelec Technical Assistance Center (TAC). Each CSR includes an individual tracking number. Together with TAC Engineers, the representative will help you resolve your request.

The Customer Care Center is available 24 hours a day, 7 days a week, 365 days a year, and is linked to TAC Engineers around the globe.

Tekelec TAC Engineers are available to provide solutions to your technical questions and issues 7 days a week, 24 hours a day. After a CSR is issued, the TAC Engineer determines the classification of the trouble. If a critical problem exists, emergency procedures are initiated. If the problem is not critical, normal support procedures apply. A primary Technical Engineer is assigned to work on the CSR and provide a solution to the problem. The CSR is closed when the problem is resolved.

Tekelec Technical Assistance Centers are located around the globe in the following locations:

Tekelec - Global

Email (All Regions): support@tekelec.com

- **USA and Canada**

  Phone:

  1-888-FOR-TKLC or 1-888-367-8552 (toll-free, within continental USA and Canada)

  1-919-460-2150 (outside continental USA and Canada)

  TAC Regional Support Office Hours:

  8:00 a.m. through 5:00 p.m. (GMT minus 5 hours), Monday through Friday, excluding holidays

- **Caribbean and Latin America (CALA)**

  Phone:

  USA access code +1-800-658-5454, then 1-888-FOR-TKLC or 1-888-367-8552 (toll-free)

  TAC Regional Support Office Hours (except Brazil):

  10:00 a.m. through 7:00 p.m. (GMT minus 6 hours), Monday through Friday, excluding holidays

  - **Argentina**

    Phone:

    0-800-555-5246 (toll-free)

  - **Brazil**

    Phone:

    0-800-891-4341 (toll-free)

    TAC Regional Support Office Hours:

    8:00 a.m. through 5:48 p.m. (GMT minus 3 hours), Monday through Friday, excluding holidays

  - **Chile**

    Phone:

    1230-020-555-5468

  - **Colombia**

    Phone:

    01-800-912-0537

  - **Dominican Republic**

    Phone:

    1-888-367-8552

  - **Mexico**

    Phone:

    001-888-367-8552

  - **Peru**

    Phone:

0800-53-087

- **Puerto Rico**

  Phone:

  1-888-367-8552 (1-888-FOR-TKLC)

- **Venezuela**

  Phone:

  0800-176-6497

- **Europe, Middle East, and Africa**

  Regional Office Hours:

  8:30 a.m. through 5:00 p.m. (GMT), Monday through Friday, excluding holidays

  - **Signaling**

    Phone:

    +44 1784 467 804 (within UK)

  - **Software Solutions**

    Phone:

    +33 3 89 33 54 00

- **Asia**

  - **India**

    Phone:

    +91 124 436 8552 or +91 124 436 8553

    TAC Regional Support Office Hours:

    10:00 a.m. through 7:00 p.m. (GMT plus 5 1/2 hours), Monday through Saturday, excluding holidays

  - **Singapore**

    Phone:

    +65 6796 2288

    TAC Regional Support Office Hours:

    9:00 a.m. through 6:00 p.m. (GMT plus 8 hours), Monday through Friday, excluding holidays

Emergency Response

In the event of a critical service situation, emergency response is offered by the Tekelec Customer Care Center 24 hours a day, 7 days a week. The emergency response provides immediate coverage, automatic escalation, and other features to ensure that the critical situation is resolved as rapidly as possible.

A critical situation is defined as a problem with the installed equipment that severely affects service, traffic, or maintenance capabilities, and requires immediate corrective action. Critical situations affect service and/or system operation resulting in one or several of these situations:

- A total system failure that results in loss of all transaction processing capability
- Significant reduction in system capacity or traffic handling capability
- Loss of the system's ability to perform automatic system reconfiguration
- Inability to restart a processor or the system
- Corruption of system databases that requires service affecting corrective actions
- Loss of access for maintenance or recovery operations
- Loss of the system ability to provide any required critical or major trouble notification

Any other problem severely affecting service, capacity/traffic, billing, and maintenance capabilities may be defined as critical by prior discussion and agreement with the Tekelec Customer Care Center.

Related Publications

The following documents provide additional information:

- *Release Notice*
- *Feature Notice*
- *Roadmap to Hardware Documentation*
- *Configuration Management Platform Wireless User's Guide*
- *Troubleshooting Guide*
- *SNMP User's Guide*
- *MPE / MRA Key Performance Indicators and Operational Measurements Application Note*
- *Multi-Protocol Routing Agent User's Guide*

Locate Product Documentation on the Customer Support Site

Access to Tekelec's Customer Support site is restricted to current Tekelec customers only. This section describes how to log into the Tekelec Customer Support site and locate a document. Viewing the document requires Adobe Acrobat Reader, which can be downloaded at www.adobe.com.

1. Log into the *Tekelec Customer Support* site.

   Note: If you have not registered for this new site, click the **Register Here** link. Have your customer number available. The response time for registration requests is 24 to 48 hours.

2. Click the **Product Support** tab.
3. Use the Search field to locate a document by its part number, release number, document name, or document type. The Search field accepts both full and partial entries.
4. Click a subject folder to browse through a list of related files.
5. To download a file to your location, right-click the file name and select **Save Target As**.

Chapter 2

Schema Definitions and Request Specifications

Topics:

To simplify integration, the OSSI XML interface uses an XML over HTTP(s) mechanism for requests and responses. Using XML as a transport allows the data to be represented in a portable, vendor-neutral and readable format.

The interface itself is divided into the following areas:

- Common Interface — defines generalized operations performed on the system.
- Topology Interface — allows users to manage and query network elements and paths.
- Subscriber Interface — allows users to manage and query subscriber elements, such as accounts and tiers.
- Operational Measurements (OM) Interface — allows users to retrieve statistical data from the system.

Each of these interfaces is described in detail in later chapters of this document.

This chapter describes the Schema Definitions and Request Specifications used within the OSSI XML interface.

Schema Definitions

The OSSI XML interface defines specific request and response messages for each of the areas outlined previously. To enable message validation and to accurately specify the syntax of each of the messages, XML schema definitions are provided as follows:

- OssiXmlRequestResponse.xsd — defines the schema for request and response formats.
- OssiXmlCommon.xsd — defines the schema for global types and requests used in all interfaces.
- OssiXmlTopology.xsd — defines the schema for the topology interface.
- OssiXmlSubscriber.xsd — defines the schema for the subscriber interface.
- OssiXmlOm.xsd — defines the schema for the OM interface.

Request Specification

Each of the requests and responses defined within the previous schema definitions are sent to and received from the Configuration Management Platform (CMP) using HTTP(s) messages. Specifically, an HTTP POST message is sent containing the specific request message. The HTTP response contains a response message indicating status and returning any data as required.

For an application desiring to use the OSSI XML interface, the only requirement is the ability to send an HTTP POST and to process any response. This guide contains examples that use the command line utility wget to send an HTTP POST request that contains data specified in an XML file as input and returns an output XML file. The example that follows demonstrates the basic wget options; additional options are available but not described here. Please note that the request URL is case sensitive and must be entered as seen here.

```
> wget --post-file=input.xml --output-document=output.xml
"http://1.2.3.4/mi/xmlInterfaceRequest.do?user=test&pwd=test"
```

Where the following describes each parameter:

- `--post-file=input.xml` (Required) — This parameter indicates the request input XML file.
- `--output-document=output.xml` (Optional) — This parameter is used to name the output file. If unspecified, the default filename is the URL string indicated in the wget request.
- `http://1.2.3.4/mi/ xmlInterfaceRequest.do ?user=test&pwd=test` (Required) — The HTTP request URL, including the authentication credentials.
- `--timeout=0` (Optional) — This parameter sets the network timeout to seconds. The default for this value is 900 (15min). A value of 0 disables timeout checking completely.
- `--progress=dot` (Optional) — This parameter is used to display the progress bar on the request.

Chapter 3

Common Responses and Commands

Topics:

This chapter describes the generic response format for many of the commands used and the commands that are common to the various OSSI interfaces.

Common Responses

This section describes the generic response format to many of the commands. The response format follows the generic response tag defined in the XSDs. The following are examples of successful and failed operations.

Example XML Response for a Successful Operation

The following is an example of an XML response for a successful operation:

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="2">Successfully imported 2 network elements.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Example XML Response for an Operation that Fails at the System Level

The following is an example of an XML response for an operation that fails at the system level (for example, a malformed request):

```
<?xml version='1.0' ?>
<Response>
    <Result>103</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="0"></Success>
        <Failure count="1">Incorrectly formatted XML. The element type
        "NetworkElementStats" must be terminated by the matching end-tag
        "</NetworkElementStats>".</Failure>
    </Command>
</Response>
```

Example XML Response for an Operation that Includes an Application-level Failure

The following is an example of an XML response for an operation that includes an application-level failure:

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="0"></Success>
        <Failure count="1">Failed to import 1 network elements.
        Network Element includes an invalid NetworkElementType.
        Network Element: RouterA</Failure>
```

```
    </Command>
</Response>
```

Result Codes

Many of the OSSI commands return a numeric result code in their response messages. This code represents the status of the operation at the system level, and shows whether the command itself could be executed or not. The following table describes the possible result codes.

Table 2: OSSI Result Codes

Result Code	Description
0	The command was valid at a system level
100	There was a general failure due to an internal server error
101	The command failed authentication; the account or password does not match between the OSSI and the URL
102	An input stream error occurred (for example, the requested data exceeds maximum size). The maximum size for a single request is 20,000,000 bytes. An example of the error message is: `errors.importExport.variableMessageMaxFileSize=Import file exceeds max size of 20M`
103	The request is malformed; for example, the XML tags may be invalid. Refer to the XSD of the OSSI query and ensure that no characters appear before the XML header (<?xml version="1.0" encoding="UTF-8"?>).

Common Commands

This section describes commands that are common to the various OSSI interfaces. These include:

- Get Version — allows users to request the OSSI API version number, as a string (for example, “2.0.3”). This enables the user to verify that the OSSI interface being used is the version expected, ensuring that the commands will operate as specified for that version.
- Distribute Updates — allows users to trigger an immediate distribution of any pending data changes to the policy servers. This enables the user to queue up many changes in the Configuration Management Platform (CMP) before sending them all at once to the policy servers in the deployment. Alternatively, changes can be distributed one-by-one on a per-command basis. The following types of data are pushed down to the policy servers as part of this request:
  - Topology data, which includes Network Elements and Paths.
  - Tiers

    - Subscriber accounts

- Serving Gateway/MCC-MNC Mapping — allows users to verify input xml files.
- Query OM Stats Setting — allows users to verify the persistent-interval settings for Reset Configuration and Collection Interval.
- Query Absolute Values for Statistics — allows users to retrieve absolute values when **Stats.Interval.Enabled.** is set to false.

Get Version

The following examples show the requests and responses that are defined in the XSDs for the GetVersion tag.

The following is an example of the request that follows the GetVersion tag.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
    <GetVersion/>
</XmlInterfaceRequest>
```

The following example response to the previous request follows the generic Response tag.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">3.1.0</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Distribute Updates

The following examples show the requests and responses that are defined in the XSDs for the DistributeUpdates tag.

The following is an example of the request that follows the DistributeUpdates tag.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
    <DistributeUpdates/>
</XmlInterfaceRequest>
```

The optional ForceSynchronization attribute can be used to force a complete resynchronization of the CMP and Multimedia Policy Engine (MPE) databases. The default value for this attribute is false.

Typically, using this attribute should not be necessary as the system should always be kept in-sync automatically; it is merely provided as a fail-safe measure for unforeseen circumstances. Also note that there is a performance impact from using this attribute, so it should not be used unless deemed necessary by Tekelec support. The following is an example of this attribute:

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
```

```
    <DistributeUpdates ForceSynchronization="true"/>
</XmlInterfaceRequest>
```

The following example response to the previous request follows the generic Response tag.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="3">Topology updates successful. Tier updates successful.
Account updates successful.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Serving Gateway/MCC-MNC Mapping

The following examples show the requests and responses that are defined in the XSDs for the SgwMapping tag.

The following is an example of the request that follows the SgwMapping tag.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<XmlInterfaceRequest>
  <AddSgwMapping>
   <SgwMapping>
    <Name>test</Name>
    <Description>description for the mcc-mnc
    mapping</Description>
    <MccMnc>111111</MccMnc>
    <SgwIpAddress>
     <IpAddress>10.0.0.129</IpAddress>
    </SgwIpAddress>
   </SgwMapping>
  </AddSgwMapping>
</XmlInterfaceRequest>
```

The following example response to the previous request follows the generic Response tag.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Successfully imported 1
 Serving Gateway/MCC-MNC Mappings.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Query OM Stats Settings

The following examples show the requests and responses that are defined in the XSDs for the QueryOmStatsSetting tag.

The following is an example of the request that follows the QueryOmStatsSetting tag.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
```

```
    <QueryOmStatsSetting/>
</XmlInterfaceRequest>
```

The following example response to the previous request follows the generic Response tag.

```
<?xml version='1.0' ?>
  <OmStatsSetting>
     <ResetConfiguration>Interval</ResetConfiguration>
     <CollectionInterval>15</CollectionInterval>
</OmStatsSetting>
```

Query Absolute Values for Statistics

The following examples show the requests and responses that are defined in the XSDs for the Statistics tag.

Note: If Stats.Interval.Enabled is set to true (meaning that Persistent Interval Statistics are in use), then the CMP does not collect absolute counter values.

The following is an example of the request that follows the Statistics tag.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
    <Statistics/>
</XmlInterfaceRequest>
```

The following example response to the previous request follows the generic Response tag.

```
<?xml version='1.0' ?>
  <Statistics>
    <DiameterAfPeerLatencyStats>
    Absolute statistics values are not available.
    </DiameterAfPeerLatencyStats>
</Statstics>
```

Chapter 4

Topology Interface

Topics:

The Topology Interface allows users to manage and query network elements, paths, and other topology-related objects within the system.

Topology Interface Requests

The Topology Interface consists of the following requests:

- Network Elements — A network element represents a node in the network, such as a router, CMTS, PDSN, B-RAS, and so on. The network elements can be organized into logical groups that subdivide the network space.
  - Add Network Element — Allows users to add new network elements and network element groups. If a network element already exists with the given identifier (Name or NeId), then this is considered an update and the new network element replaces the existing one. All relationships between that network element and other objects in the system will be maintained.
  - Update Network Element — Allows users to update existing network elements and network element groups. If no object exists with the given identifier (Name or NeId), then this is a failure condition and the object is not saved.
  - Add Network Element To Group — Allows users to associate a network element with a group.
  - Remove Network Element From Group — Allows users to remove a network element from within a group.
  - Delete Network Element — Allows users to delete a network element.
  - Delete Network Element Group — Allows users to delete a network element group.
  - Query Network Element — Allows users to query all network elements or a specific network element, returning their configured attributes.
- Paths — A path represents a route through the network, and is identified by source and destination network elements.
  - Add Path — Allows users to add new paths to the system. If a path already exists with the given identifier (Name), then this is considered an update and the new path replacing the existing one. All relationships between that path and other objects in the system are maintained.
  - Update Path — Allows users to update paths within the system. If no object exists with the given identifier (Name), then this is a failure condition and the object is not saved.
  - Delete Path — Removes a path from the system.
  - Query Path — Allows users to query a specific path or list of paths, returning their configured attributes.
- Applications — An application is a service in the network (for example, VoIP) for which quality of service is managed.
  - AddApplication — Allows users to add new applications to the system. If an application already exists with the given identifier (Name), then this is considered an update and the new application replaces the existing one. All relationships between that application and other objects in the system are maintained.
  - UpdateApplication — Allows users to update application within the system. If no object exists with the given identifier (Name), then this is a failure condition and the object is not saved.
  - DeleteApplication — Removes an application from the system.
  - QueryApplication — Allows users to query a specific application orl ist of traffic profiles, returning their configured attributes.
- Traffic Profiles — A traffic profile is a set of parameters and their values, which are used in protocol messages to define quality of service for sessions.

  - AddTrafficProfile — Allows users to add new traffic profiles to the system. If a traffic profile already exists with the given identifier (Name), then this is considered an update and the new traffic profile replaces the existing one. All relationships between that traffic profile and other objects in the system are maintained.
  - UpdateTrafficProfile — Allows users to update traffic profiles within the system. If no object exists with the given identifier (Name), then this is a failure condition and the object is not saved.
  - DeleteTrafficProfile — Removes a traffic profile from the system.
  - QueryTrafficProfile — Allows users to query a specific traffic
  - profile or list of traffic profiles, returning their configured
  - attributes.

- Time Periods — A time period defines a set of related time slots, which receives the same quality of service (for example, "the peak time of network usage is Monday through Friday from 9:00 am to 5:00 pm").
  - AddTimePeriod — Allows users to add new time periods to the system. If a time period already exists with the given identifier (Name), then this is considered an update and the new time period replaces the existing one. All relationships between that time period and other objects in the system are maintained.
  - UpdateTimePeriod — Allows users to update time periods within the system. If no object exists with the given identifier (Name), then this is a failure condition and the object is not saved.
  - DeleteTimePeriod — Removes a time period from the system.
  - QueryTimePeriod — Allows users to query a specific time period or list of time periods, returning their configured attributes.

- Charging Servers — A charging server is a network element that processes accounting information for billing purposes.
  - AddChargingServer — Allows users to add new charging servers to the system. If a charging server already exists with the given identifier (Name), then this is considered an update and the new charging server replaces the existing one. All relationships between that charging server and other objects in the system are maintained.
  - UpdateChargingServer — Allows users to update charging servers within the system. If no object exists with the given identifier (Name), then this is a failure condition and the object is not saved.
  - DeleteChargingServer — Removes a charging server from the system.
  - QueryChargingServer — Allows users to query a specific charging server or list of charging servers, returning their configured attributes.

- Match List — A match list is a set of values that can be used within one ore more policy definitions.
  - AddMatchList — Allows users to add new match list to the system.If a match list already exists with the given identifier (Name), then this is considered an update and the new match list replaces the existing one. All relationships between that match list and other objects in the system are maintained.
  - UpdateMatchList — Allows users to update match lists within the system. If no object exists with the given identifier (Name), then this is a failure condition and the object is not saved.
  - DeleteMatchList — Removes a match list from the system.
  - QueryMatchList — Allows users to query a specific match list or list of match lists, returning their configured attributes.

Network Elements

A network element represents a node in the network, such as a router, CMTS, PDSN, B-RAS, and so on. The network elements can be organized into logical groups that subdivide the network space. The following sections describe the available network tags.

Add Network Element

The following examples show the request and response that are defined in the XSDs for the AddNetworkElement tag.

Request

This request follows the AddNetworkElement tag defined in the XSDs.

The following example creates a new Network Element to represent a router in the network. It also creates a group, and puts the router in that group.

The router contains the following attributes:

- Name (unique identifier): Router 23
- Description: Core router for the north east
- HostName (IP address or DNS hostname): 12.1.1.1
- NE-ID (another unique identifier): 112222
- Capacity (aggregate capacity in bps): 456000

The group contains the following attributes:

- Name: NE Group 1
- Description: All core routers

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
<AddNetworkElement>
    <NetworkElement>
        <Name>Router 23</Name>
        <Description>Core router for the north east</Description>
        <HostName>12.1.1.1</HostName>
        <NeId>112222</NeId>

  <NetworkElementType>Router</NetworkElementType>
        <NeSubType></NeSubType>
        <Capacity>456000</Capacity>
    </NetworkElement>
    <NetworkElementGroup>
        <Name>NE Group 1</Name>
        <Description>All core routers</Description>
        <RootGroup>true</RootGroup>
        <ElementRef>
            <Name>router 23</Name>
            <SubGroup>false</SubGroup>
        </ElementRef>
    </NetworkElementGroup>
</AddNetworkElement>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

The example XML response that follows represents a successful operation that creates a new network element and/or groups:

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="2">Successfully imported 1 network elements.
        Successfully imported 1 group.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

The example XML response that follows represents an operation that includes both a success and a failure (for example, an attempt to update a group's description field (success), and add an unknown network element "router 24" to the group "NE Group 1" (failed)).

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Successfully updated 1 groups.</Success>
        <Failure count="1">Failed to update 1 groups.  Network Element Group
update returned an error attempting to add the following: router 24
Group NE Group 1</Failure>
    </Command>
</Response>
```

Update Network Element

The following examples show the request and response that are defined in the XSDs for the UpdateNetworkElement tag.

Request

This request follows the UpdateNetworkElement tag defined in the XSDs.

The following example changes the capacity attribute of the network element named "Router 23" to the new value: 4567000.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
   <UpdateNetworkElement>
       <NetworkElement>
           <Name>Router 23</Name>
           <Capacity>4567000</Capacity>
       </NetworkElement>
   </UpdateNetworkElement>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
```

```
        <Success count="1">Successfully updated 1 network elements.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Add Network Element to Group

The following examples show the request and response that are defined in the XSDs for the AddNetworkElementToGroup tag.

Request

This request follows the AddNetworkElementToGroup tag defined in the XSDs.

The following example adds three network elements (Router 2, Router 34, and Router 131) to a group (Group 1).

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
   <AddNetworkElementToGroup>
       <GroupName>Group 1</GroupName>
       <Name>Router 2</Name>
       <Name>Router 34</Name>
       <Name>Router 131</Name>
   </AddNetworkElementToGroup>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="3">Added 3 elements to a group.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Remove Network Element from Group

The following examples show the request and response that are defined in the XSDs for the RemoveNetworkElementFromGroup tag.

Request

This request follows the RemoveNetworkElementFromGroup tag defined in the XSDs.

The following example removes two network elements (Router 34 and Router 131) from a group (Group 1).

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
   <RemoveNetworkElementFromGroup>
       <GroupName>Group 1</GroupName>
       <Name>Router 34</Name>
       <Name>Router 131</Name>
```

```
   </RemoveNetworkElementFromGroup>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="2">Removed 2 elements from group.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Delete Network Element

The following examples show the request and response that are defined in the XSDs for the DeleteNetworkElement tag.

Request

This request follows the DeleteNetworkElement tag defined in the XSDs.

The following example deletes two network elements (Node 1 and Node 2).

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
   <DeleteNetworkElement>
      <Name>Node1</Name>
      <Name>Node2</Name>
   </DeleteNetworkElement>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="2">Deleted 2 elements.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Delete Network Element Group

The following examples show the request and response that are defined in the XSDs for the DeleteNetworkElementGroup tag.

Request

This request follows the DeleteNetworkElementGroup tag defined in the XSDs.

The following example deletes two network element groups (Group 1 and Group 2).

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
<DeleteNetworkElementGroup>
    <Name>Group 1</Name>
    <Name>Group 2</Name>
</DeleteNetworkElementGroup>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="2">Deleted 2 network element groups.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Query Network Elements

The following examples show the request and response that are defined in the XSDs for the QueryNetworkElement tag.

Request

This request follows the QueryNetworkElement tag defined in the XSDs.

Example 1 — XML for querying multiple network elements by name.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
    <QueryNetworkElement>
        <Name>CMTS A</Name>
        <Name>CMTS B</Name>
</QueryNetworkElement>
</XmlInterfaceRequest>
```

Example 2 — XML for querying all network elements and groups.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
   <QueryNetworkElement/>
</XmlInterfaceRequest>
```

Response

The response to this request follows the ConfigurationData -> NetworkElement tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="">
    <NetworkElement>
        <Name>CMTS A</Name>
        <Description></Description>
        <HostName>10.0.2.6</HostName>
```

```
        <BackupHostName></BackupHostName>
        <NeId></NeId>

<NetworkElementType>Router</NetworkElementType>
        <NeSubType></NeSubType>
        <Managed>true</Managed>
        <PcmmPort>3918</PcmmPort>
        <DqosPort>2126</DqosPort>
        <ReadCommunity>public</ReadCommunity>
        <Capacity>0</Capacity>
        <X>0</X>
        <Y>0</Y>
        <AdminState>0</AdminState>
        <LearnedSubnet>
            <IpAddress>10.128.3.0</IpAddress>
            <Mask>24</Mask>
        </LearnedSubnet>
        <LearnedSubnet>
            <IpAddress>10.254.3.0</IpAddress>
            <Mask>24</Mask>
        </LearnedSubnet>
    </NetworkElement>
    <NetworkElement>
        <Name>CMTS B</Name>
        <Description></Description>
        <HostName>10.0.3.2</HostName>
        <BackupHostName></BackupHostName>
        <NeId></NeId>

<NetworkElementType>Router</NetworkElementType>
        <NeSubType></NeSubType>
        <Managed>true</Managed>
        <PcmmPort>3918</PcmmPort>
        <DqosPort>2126</DqosPort>
        <ReadCommunity>public</ReadCommunity>
        <Capacity>0</Capacity>
        <X>0</X>
        <Y>0</Y>
        <AdminState>0</AdminState>
        <LearnedSubnet>
            <IpAddress>10.128.1.0</IpAddress>
            <Mask>24</Mask>
        </LearnedSubnet>
        <LearnedSubnet>
            <IpAddress>10.254.1.0</IpAddress>
            <Mask>24</Mask>
        </LearnedSubnet>
        <LearnedSubnet>
            <IpAddress>11.129.0.0</IpAddress>
            <Mask>16</Mask>
        </LearnedSubnet>
    </NetworkElement>
</ConfigurationData>
```

Paths

A path represents a route through the network, and is identified by source and destination network elements.

Add Path

The following examples show the request and response that are defined in the XSDs for the AddPath tag.

Request

This request follows the AddPath tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
<AddPath>
    <Path>
        <Name>Path1</Name>
        <Description/>
        <Hops>
            <Hop>
                <NeName>arris1</NeName>
            </Hop>
            <Hop>
                <NeName>cisco1</NeName>
            </Hop>
            <Hop>
                <NeName>moto1</NeName>
            </Hop>
        </Hops>
    </Path>
    <Path>
        <Name>Path2</Name>
        <Description/>
        <Hops>
            <Hop>
                <NeName>moto2</NeName>
            </Hop>
            <Hop>
                <NeName>cisco2</NeName>
            </Hop>
            <Hop>
                <NeName>arris2</NeName>
            </Hop>
        </Hops>
    </Path>
    <Path>
        <Name>Path3</Name>
        <Description/>
        <Hops>
            <Hop>
                <NeName>arris3</NeName>
            </Hop>
            <Hop>
                <NeName>moto3</NeName>
            </Hop>
        </Hops>
    </Path>
</AddPath>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
```

```
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="3">Successfully imported 3 paths.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Update Path

The following examples show the request and response that are defined in the XSDs for the UpdatePath tag.

Request

This request follows the UpdatePath tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
<UpdatePath>
    <Path>
        <Name>Path1</Name>
        <Hops>
            <Hop>
                <NeName>moto1</NeName>
            </Hop>
            <Hop>
                <NeName>cisco1</NeName>
            </Hop>
            <Hop>
                <NeName>arris2</NeName>
            </Hop>
        </Hops>
    </Path>
 </UpdatePath>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Successfully updated 1 paths.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Delete Path

The following examples show the request and response that are defined in the XSDs for the DeletePath tag.

Request

This request follows the DeletePath tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<XmlInterfaceRequest>
<DeletePath>
    <Name>Path1</Name>
    <Name>Path2</Name>
</DeletePath>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="2">Deleted 2 paths.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Query Path

The following examples show the request and response that are defined in the XSDs for the QueryPath tag.

Request

This request follows the QueryPath tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
    <QueryPath>
        <Name>Path1</Name>
        <Name>Path2</Name>
</QueryPath>
</XmlInterfaceRequest>
```

Response

The response to this request follows the ConfigurationData -> Path tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="3.0.0">
    <Path>
        <Name>Path1</Name>
        <Description/>
        <Source>
            <Address>1.1.1.2</Address>
            <Mask>24</Mask>
        </Source>
        <Destination>
            <Address>3.3.3.2</Address>
            <Mask>24</Mask>
        </Destination>
        <Hops>
            <Hop>
                <NeName>arris</NeName>
            </Hop>
            <Hop>
                <NeName>cisco</NeName>
            </Hop>
```

```
            <Hop>
                <NeName>moto</NeName>
            </Hop>
        </Hops>
    </Path>
    <Path>
        <Name>Path2</Name>
        <Description/>
        <Source>
            <Address>3.3.3.2</Address>
            <Mask>24</Mask>
        </Source>
        <Destination>
            <Address>1.1.1.2</Address>
            <Mask>24</Mask>
        </Destination>
        <Hops>
            <Hop>
                <NeName>moto</NeName>
            </Hop>
            <Hop>
                <NeName>cisco</NeName>
            </Hop>
            <Hop>
                <NeName>arris</NeName>
            </Hop>
        </Hops>
    </Path>
</ConfigurationData>
```

Applications

An application is a service in the network (for example, VoIP) for which Quality of Service (QoS) is managed.

Add Application

The following examples show the request and response that are defined in the XSDs for the AddApplication tag.

Request

This request follows the AddApplication tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <AddApplication>
        <Application>
            <Name>VoD</Name>
            <Description>Video on Demand</Description>
            <LatencySensitive>true</LatencySensitive>
            <Tracked>true</Tracked>
            <UpGatesPerSession>1</UpGatesPerSession>
            <DownGatesPerSession>1</DownGatesPerSession>
            <TrackingTimeoutTime>240</TrackingTimeoutTime>
            <TrackingTimeoutUnit>1</TrackingTimeoutUnit>
        </Application>
```

```
    </AddApplication>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Successfully imported 1 applications.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Update Application

The following examples show the request and response that are defined in the XSDs for the UpdateApplication tag.

Request

This request follows the UpdateApplication tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <UpdateApplication>
        <Application>
            <Name>VoD</Name>
            <Description>Video on Demand</Description>
            <LatencySensitive>false</LatencySensitive>
            <UpGatesPerSession>3</UpGatesPerSession>
        </Application>
    </UpdateApplication>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Successfully updated 1 applications.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Delete Application

The following examples show the request and response that are defined in the XSDs for the DeleteApplication tag.

Request

This request follows the DeleteApplication tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <DeleteApplication>
      <Name>VoD</Name>
    </DeleteApplication>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Deleted 1 applications.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Query Application

The following examples show the request and response that are defined in the XSDs for the QueryApplication tag.

Request

This request follows the QueryApplication tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <QueryApplication/>
</XmlInterfaceRequest>
```

Response

The response to this request follows the ConfigurationData -> Application tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="">
   <Application>
        <Name>VoD</Name>
        <Description>Video on Demand</Description>
        <LatencySensitive>true</LatencySensitive>
        <Tracked>true</Tracked>
        <UpGatesPerSession>1</UpGatesPerSession>
        <DownGatesPerSession>1</DownGatesPerSession>
        <TrackingTimeoutTime>240</TrackingTimeoutTime>
        <TrackingTimeoutUnit>1</TrackingTimeoutUnit>
```

```
   </Application>
</ConfigurationData>
```

Traffic Profiles

A traffic profile is a set of parameters and their values, which are used in protocol messages to define QoS for sessions.

Add Traffic Profile

The following examples show the request and response that are defined in the XSDs for the AddTrafficProfile tag.

Request

This request follows the AddTrafficProfile tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <AddTrafficProfile>
        <QosProfile>
            <Name>default-rule-traffic-profile</Name>
            <ProfileType>0</ProfileType>
            <QosProfileType>Predefined PCC Rule</QosProfileType>
            <QosProfileProfileParam>
            <ProfileParamUid>param.diameter. pccPredefRule.profileTypeNumber
              </ProfileParamUid>
            <ParamValue>11</ParamValue>
            </QosProfileProfileParam>
            <QosProfileProfileParam>
            <ProfileParamUid>param.diameter. pccPredefRule.name
              </ProfileParamUid>
            <ParamValue>PCRF-DEFAULT</ParamValue>
            </QosProfileProfileParam>
            <QosProfileProfileParam>
            <ProfileParamUid>param.diameter. pccPredefRule.description
              </ProfileParamUid>
            <ParamValue>This is the static rule that gets applied
              by default</ParamValue>
            </QosProfileProfileParam>
        </QosProfile>
    </AddTrafficProfile>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Successfully imported 1 Traffic Profile(s).</Success>
        <Failure count="0"></Failure>
```

```
    </Command>
</Response>
```

Update Traffic Profile

The following examples show the request and response that are defined in the XSDs for the UpdateTrafficProfile tag.

Request

This request follows the UpdateTrafficProfile tag defined in the XSDs.

```
<<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <UpdateTrafficProfile>
        <QosProfile>
            <Name>default-rule-traffic-profile</Name>
            <ProfileType>0</ProfileType>
            <QosProfileType>Second Predefined PCC Rule</QosProfileType>
        </QosProfile>
   </UpdateTrafficProfile>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Successfully updated 1 Traffic Profile(s).</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Delete Traffic Profile

The following examples show the request and response that are defined in the XSDs for the DeleteTrafficProfile tag.

Request

This request follows the DeleteTrafficProfile tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <DeleteTrafficProfile>
      <Name>default-rule-traffic-profile</Name>
    </DeleteTrafficProfile>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
```

```
    <Command type="XmlInterfaceResponse">
        <Success count="1">Deleted 1 Traffic Profile(s).</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Query Traffic Profile

The following examples show the request and response that are defined in the XSDs for the QueryTrafficProfile tag that queries all traffic profiles:

Request

This request follows the QueryTrafficProfile tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <QueryTrafficProfile/>
</XmlInterfaceRequest>
```

Response

The response to this request follows the ConfigurationData -> QosProfile tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="">
        <QosProfile>
            <Name>default-rule-traffic-profile</Name>
            <ProfileType>0</ProfileType>
            <QosProfileType>Predefined PCC Rule</QosProfileType>
            <QosProfileProfileParam>
            <ProfileParamUid>param.diameter.pccPredefRule.profileTypeNumber
            </ProfileParamUid>
            <ParamValue>11</ParamValue>
            </QosProfileProfileParam>
            <QosProfileProfileParam>
            <ProfileParamUid>param.diameter.pccPredefRule.name
              </ProfileParamUid>
            <ParamValue>PCRF-DEFAULT</ParamValue>
            </QosProfileProfileParam>
            <QosProfileProfileParam>
            <ProfileParamUid>param.diameter.pccPredefRule.description
              </ProfileParamUid>
            <ParamValue>This is the static rule that gets applied by default
              </ParamValue>
            </QosProfileProfileParam>
        </QosProfile>
        <QosProfile>
            <Name>p2p-rule-traffic-profile</Name>
            <ProfileType>0</ProfileType>
            <QosProfileType>Predefined PCC Rule</QosProfileType>
            <QosProfileProfileParam>
            <ProfileParamUid>param.diameter.pccPredefRule.profileTypeNumber
              </ProfileParamUid>
            <ParamValue>11</ParamValue>
            </QosProfileProfileParam>
            <QosProfileProfileParam>
            <ProfileParamUid>param.diameter.pccPredefRule.name </ProfileParamUid>
            <ParamValue>P2P-RATE-LIMIT-64K</ParamValue>
            </QosProfileProfileParam>
            <QosProfileProfileParam>
            <ProfileParamUid>param.diameter.pccPredefRule.description
```

```
</ProfileParamUid>
           <ParamValue>Rule limits bit-torrent and other peer to peer applications

              to 64Kbps downstream</ParamValue>
            </QosProfileProfileParam>
     </QosProfile>
</ConfigurationData>
```

Time Periods

A time period defines a set of related time slots, which receives the same QoS (for example, “the peak time of network usage is Monday through Friday from 9:00 am to 5:00 pm”).

Add Time Period

The following examples show the request and response that are defined in the XSDs for the AddTimePeriod tag.

Request

This request follows the AddTimePeriod tag defined in the XSDs.

The StartTime and EndTime tags represent a time-of-day value expressed in milliseconds. For example, “01:30” time-of-day corresponds to 5,400,000 ((60+30)*60*1000) when expressed in milliseconds.

The Mask tag represents the day-of-week expressed in the form of a bitmask. The value for each day-of-week corresponds to a power of two: Sunday is 1, Monday is 2, Tuesday is 4, Wednesday is 8, Thursday is 16, Friday is 32, and Saturday is 64. To specify a set of days, add together the values associated with the respective days. For example, "Monday and Wednesday" corresponds to a value of 10 (i.e. 2 + 8).

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
   <AddTimePeriod>
        <TimePeriod>
            <Name>Weekday Peak</Name>
            <Description></Description>
            <Precedence>1</Precedence>
            <TimeSlot>
                <Mask>32</Mask>
                <StartTime>24300000</StartTime>
                <EndTime>39600000</EndTime>
                <StartDate></StartDate>
                <EndDate></EndDate>
            </TimeSlot>
        </TimePeriod>
    </AddTimePeriod>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
```

```
        <Success count="1">Successfully imported 1 Time Period(s).</Success>
         <Failure count="0"></Failure>
    </Command>
</Response>
```

Update Time Period

The following examples show the request and response that are defined in the XSDs for the UpdateTimePeriod tag.

Request

This request follows the UpdateTimePeriod tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
   <UpdateTimePeriod>
      <TimePeriod>
            <Name>Weekday Peak</Name>
            <Description></Description>
            <Precedence>44</Precedence>
            <TimeSlot>
                <Mask>32</Mask>
                <StartTime>24300000</StartTime>
                <EndTime>39600000</EndTime>
                <StartDate></StartDate>
                <EndDate></EndDate>
            </TimeSlot>
        </TimePeriod>
   </UpdateTimePeriod>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Successfully updated 1 Time Period(s).</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Delete Time Period

The following examples show the request and response that are defined in the XSDs for the DeleteTimePeriod tag.

Request

This request follows the DeleteTimePeriod tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <DeleteTimePeriod>
      <Name>Weekday Peak/Name>
```

```
    </DeleteTimePeriod>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Deleted 1 Time Period(s).</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Query Time Period

The following examples show the request and response that are defined in the XSDs for the QueryTimePeriod tag.

Request

This request follows the QueryTimePeriod tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <QueryTimePeriod/>
</XmlInterfaceRequest>
```

Response

The response to this request follows the ConfigurationData -> TimePeriod tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="">
    <TimePeriod>
        <Name>Weekday Peak</Name>
        <Description></Description>
        <Precedence>1</Precedence>
        <TimeSlot>
            <Mask>32</Mask>
            <StartTime>24300000</StartTime>
            <EndTime>39600000</EndTime>
            <StartDate></StartDate>
            <EndDate></EndDate>
        </TimeSlot>
    </TimePeriod>
    <TimePeriod>
        <Name>Weekend Peak</Name>
        <Description></Description>
        <Precedence>7</Precedence>
        <TimeSlot>
            <Mask>32</Mask>
            <StartTime>25200000</StartTime>
            <EndTime>37800000</EndTime>
            <StartDate></StartDate>
            <EndDate></EndDate>
        </TimeSlot>
    </TimePeriod>
    <TimePeriod>
```

```
        <Name>Weekend Offpeak</Name>
        <Description></Description>
        <Precedence>0</Precedence>
        <TimeSlot>
            <Mask>32</Mask>
            <StartTime>23400000</StartTime>
            <EndTime>39600000</EndTime>
            <StartDate></StartDate>
            <EndDate></EndDate>
        </TimeSlot>
    </TimePeriod>
</ConfigurationData>
```

Charging Servers

A charging server is a network element that processes accounting information for billing purposes.

Add Charging Server

The following examples show the request and response that are defined in the XSDs for the AddChargingServer tag.

Request

This request follows the AddChargingServer tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <AddChargingServer>
        <ChargingServer>
        <Name>chargingServer_import1</Name>
        <Description></Description>
        <Security>true</Security>
        <HostName>80.20.20.101</HostName>
        <Port>4040</Port>
        <Transport>udp</Transport>
        <Protocol>radius</Protocol>
    </ChargingServer>
    <ChargingServer>
        <Name>chargingServer_import2</Name>
        <Description></Description>
        <Security>false</Security>
        <HostName>12.6.43.201</HostName>
        <Port>7089</Port>
        <Transport>tcp</Transport>
        <Protocol>tacacs+</Protocol>
    </ChargingServer>
    </AddChargingServer>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
```

```
        <Success count="2">Successfully imported 2 Charging Server(s).</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Update Charging Server

The following examples show the request and response that are defined in the XSDs for the UpdateChargingServer tag.

Request

This request follows the UpdateChargingServer tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <UpdateChargingServer>
      <ChargingServer>
        <Name>chargingServer_3</Name>
        <Description></Description>
        <Security>true</Security>
        <HostName>80.20.20.100</HostName>
        <Port>6040</Port>
        <Transport>tcp</Transport>
        <Protocol>radius</Protocol>
      </ChargingServer>
    </UpdateChargingServer>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Successfully updated 1 Charging Server(s).</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Delete Charging Server

The following examples show the request and response that are defined in the XSDs for the DeleteChargingServer tag.

Request

This request follows the DeleteChargingServer tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <DeleteChargingServer>
      <Name>chs1</Name>
    </DeleteChargingServer>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="2">Deleted 2 Charging Server(s).</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Query Charging Server

The following examples show the request and response that are defined in the XSDs for the QueryChargingServer tag.

Request

This request follows the QueryChargingServer tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
    <QueryChargingServer>
        <Name>chargingServer_3</Name>
        <Name>chargingServer_4</Name>
        <Name>chargingServer_import1</Name>
        <Name>chargingServer_import2</Name>
    </QueryChargingServer>
</XmlInterfaceRequest>
```

Response

The response to this request follows the ConfigurationData -> ChargingServer tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="">
    <ChargingServer>
        <Name>chargingServer_3</Name>
        <Description></Description>
        <Security>true</Security>
        <HostName>80.20.20.100</HostName>
        <Port>4040</Port>
        <Transport>udp</Transport>
        <Protocol>radius</Protocol>
    </ChargingServer>
    <ChargingServer>
        <Name>chargingServer_4</Name>
        <Description></Description>
        <Security>false</Security>
        <HostName>12.6.43.200</HostName>
        <Port>7089</Port>
        <Transport>tcp</Transport>
        <Protocol>tacacs+</Protocol>
    </ChargingServer>
    <ChargingServer>
        <Name>chargingServer_import1</Name>
        <Description></Description>
        <Security>true</Security>
        <HostName>80.20.20.101</HostName>
        <Port>4040</Port>
        <Transport>udp</Transport>
```

```
        <Protocol>radius</Protocol>
    </ChargingServer>
    <ChargingServer>
        <Name>chargingServer_import2</Name>
        <Description></Description>
        <Security>false</Security>
        <HostName>12.6.43.201</HostName>
        <Port>7089</Port>
        <Transport>tcp</Transport>
        <Protocol>tacacs+</Protocol>
    </ChargingServer>
</ConfigurationData>
```

Match List

The match list is a set of values that can be used within one or more policy definitions. When the user writes a policy rule to compare against a value in a list, the list or set of lists to be used for comparison is explicitly specified. For example:

"Where the SGSNIpAddress is contained in MaltaSgsnList, GreekSgsnList"

The Match List Type determines what kind of values can be specified in that list and how those values are matched within the policy rule. Types can be "IPv4 Address" (allows address and subnet), "IPv6 Address"(allows address and prefix), "String", and "Wildcard".

Add Match List

The following examples show the request and response that are defined in the XSDs for the AddMatchList tag.

Request

This request follows the AddMatchList tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
<AddMatchList>
  <TokenList>
      <Name>match_list_01a_sgsn</Name>
      <Description>This is a match list that will match the given
        SGSN</Description>
    <ItemType>3</ItemType>
    <TokenListItem>
      <Value>44.44.44.44</Value>
    </TokenListItem>
    <TokenListItem>
      <Value>55.55.55.55</Value>
    </TokenListItem>
    <TokenListItem>
      <Value>66.66.66.66</Value>
    </TokenListItem>
  </TokenList>
  <TokenList>
    <Name>match_list_01b_sgsn_black</Name>
    <Description />
    <ItemType>1</ItemType>
    <TokenListItem>
      <Value>44.44.44.44</Value>
```

```
    </TokenListItem>
    <TokenListItem>
      <Value>55.55.55.55</Value>
    </TokenListItem>
    <TokenListItem>
      <Value>66.66.66.66</Value>
    </TokenListItem>
  </TokenList>
</AddMatchList>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="2">Successfully imported 2 Match List(s).</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Update Match List

The following examples show the request and response that are defined in the XSDs for the UpdateMatchList tag.

Request

This request follows the UpdateMatchList tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
<UpdateMatchList>
  <TokenList>
      <Name>match_list_01a_sgsn</Name>
      <Description>This is an updated match list that will match the
        given SGSN</Description>
    <ItemType>3</ItemType>
    <TokenListItem>
      <Value>44.44.44.77</Value>
    </TokenListItem>
    <TokenListItem>
     <Value>66.66.66.77</Value>
    </TokenListItem>
  </TokenList>
</UpdateMatchList>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Successfully updated 1 Match List(s).</Success>
         <Failure count="0"></Failure>
```

```
    </Command>
</Response>
```

Delete Match List

The following examples show the request and response that are defined in the XSDs for the DeleteMatchList tag.

Request

This request follows the DeleteMatchList tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <DeleteMatchList>
      <Name>match_list_01a_sgsn</Name>
    </DeleteMatchList>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="2">Deleted 1 Match List(s).</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Query Match List

The following examples show the request and response that are defined in the XSDs for the QueryMatchList tag.

Request

This request follows the QueryMatchList tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
    <QueryMatchList>
        <Name>match_list_01a_sgsn</Name>
        <Name>match_list_01b_sgsn_black</Name>
</QueryMatchList>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="">
  <TokenList>
      <Name>match_list_01a_sgsn</Name>
      <Description>This is a match list that will match the given
        SGSN</Description>
```

```
    <ItemType>3</ItemType>
    <TokenListItem>
      <Value>44.44.44.77</Value>
    </TokenListItem>
    <TokenListItem>
      <Value>66.66.66.77</Value>
    </TokenListItem>
  </TokenList>
  <TokenList>
    <Name>match_list_01b_sgsn_black</Name>
    <Description />
    <ItemType>1</ItemType>
    <TokenListItem>
      <Value>44.44.44.44</Value>
    </TokenListItem>
    <TokenListItem>
      <Value>55.55.55.55</Value>
    </TokenListItem>
    <TokenListItem>
      <Value>66.66.66.66</Value>
    </TokenListItem>
  </TokenList>
</ConfigurationData>
```

Monitoring Key

A monitoring key is a unique character string that identifies the quota profile to be used by a policy and charging control (PCC) rule for usage tracking. The monitoring key is associated with the quota profile by selecting a policy action that grants usage to a selected number of quota profiles.

Add Monitoring Key

The following examples show the request and response that are defined in the XSDs for the AddMonitoringKey tag.

Request

This request follows the AddMonitoringKey tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <AddMonitoringKey>
        <MonitoringKey>
            <Name>Monitoring Key 1</Name>
            <Description>Monitoring Key 1</Description>
            <MonitoringType>0</MonitoringType>
            <MonitoringKey>KEY_1</MonitoringKey>
        </MonitoringKey>
        <MonitoringKey>
            <Name>Monitoring Key 2</Name>
            <Description></Description>
            <MonitoringType>2</MonitoringType>
            <MonitoringKey>KEY_2</MonitoringKey>
        </MonitoringKey>
    </AddMonitoringKey>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
     <Result>0</Result>
     <Command type="XmlInterfaceResponse">
            <Success count="2">Successfully imported 2 monitoringKey(s).</Success>

             <Failure count="0"></Failure>
     </Command>
</Response>
```

Update Monitoring Key

The following examples show the request and response that are defined in the XSDs for the UpdateMonitoringKey tag.

Request

This request follows the UpdateMonitoringKey tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
<UpdateMonitoringKey>
        <MonitoringKey>
            <Name>Monitoring Key 2</Name>
            <Description></Description>
            <MonitoringType>2</MonitoringType>
            <MonitoringKey>KEY_2</MonitoringKey>
        </MonitoringKey>
    </UpdateMonitoringKey>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response> <Result>0</Result>
 <Command type="XmlInterfaceResponse">
  <Success count="1">Successfully updated 1 monitoringKey(s).</Success>
   <Failure count="0"></Failure>
 </Command>
</Response>
```

Delete Monitoring Key

The following examples show the request and response that are defined in the XSDs for the DeleteMonitoringKey tag.

Request

This request follows the DeleteMonitoringKey tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
 <DeleteMonitoringKey>
  <MonitoringKey>
   <Name>Monitoring Key 2</Name>
```

```
  </MonitoringKey>
 </DeleteMonitoringKey>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
        <Result>0</Result>
        <Command type="XmlInterfaceResponse">
                <Success count="1">Deleted 1 Monitoring Keys.</Success>
                <Failure count="0"></Failure>
        </Command>
</Response>
```

Query Monitoring Key

The following examples show the request and response that are defined in the XSDs for the QueryMonitoringKey tag.

Request

This request follows the QueryMonitoringKey tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
 <QueryChargingServer>
   <Name>chargingServer_3</Name>
   <Name>chargingServer_4</Name>
   <Name>chargingServer_import1</Name>
   <Name>chargingServer_import2</Name>
 </QueryChargingServer>
</XmlInterfaceRequest>
```

Response

The response to this request follows the **ConfigurationData ➤ MonitoringKey** tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="8.0.0">
    <MonitoringKey>
        <Name>Monitoring Key 1</Name>
        <Description>Monitoring Key 1</Description>
        <MonitoringType>0</MonitoringType>
        <MonitoringKey>KEY_1</MonitoringKey>
    </MonitoringKey>
</ConfigurationData>
```

AVP Definition

An AVP is a Diameter protocol used to encapsulate protocol-specific information with usage monitoring supported by the MPE. Diameter messages such as AAR, CCA, CCR, and RAR, are supported by 3rd

Party AVP policy conditions. The supported outgoing messages set or remove 3rd Party AVPs in Diameter.

Add AVP Definition

The following examples show the request and response that are defined in the XSDs for the AddAvpDefinition tag.

Request

This request follows the AddAvpDefinition tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <AddAvpDefinition>
        <AvpDefinition>
            <Name>AVPSAMPLE:1123</Name>
            <Description></Description>
            <AvpName>AVPSAMPLE</AvpName>
            <AvpCode>0</AvpCode>
            <VendorId>1123</VendorId>
            <MandatoryFlag>false</MandatoryFlag>
            <ProtectFlag>false</ProtectFlag>
            <MayencryptFlag>false</MayencryptFlag>
            <VendorSpecificFlag>true</VendorSpecificFlag>
            <AvpType>float32</AvpType>
            <RootAvp></RootAvp>
        </AvpDefinition>
    </AddAvpDefinition>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
   <Result>0</Result>
   <Command type="XmlInterfaceResponse">
       <Success count="1">Successfully imported 1 AVP Definition(s).</Success>
       <Failure count="0"></Failure>
   </Command>
</Response>
```

Update AVP Definition

The following examples show the request and response that are defined in the XSDs for the UpdateAvpDefinition tag.

Request

This request follows the UpdateAvpDefinition tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <UpdateAvpDefinition>
        <AvpDefinition>
            <Name>AVPSAMPLE:1123</Name>
            <Description></Description>
            <AvpName>AVPSAMPLE</AvpName>
```

```
            <AvpCode>567</AvpCode>
            <VendorId>1123</VendorId>
            <MandatoryFlag>false</MandatoryFlag>
            <ProtectFlag>false</ProtectFlag>
            <MayencryptFlag>false</MayencryptFlag>
            <VendorSpecificFlag>true</VendorSpecificFlag>
            <AvpType>float32</AvpType>
            <RootAvp></RootAvp>
        </AvpDefinition>
    </UpdateAvpDefinition>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Successfully imported 1 AVP Definition(s).</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Delete AVP Definition

The following examples show the request and response that are defined in the XSDs for the DeleteAvpDefinition tag.

Request

This request follows the DeleteAvpDefinition tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <DeleteAvpDefinition>
        <AvpDefinition>
            <Name>AVPSAMPLE:1123</Name>
        </AvpDefinition>
    </DeleteAvpDefinition>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
        <Result>0</Result>
        <Command type="XmlInterfaceResponse">
                <Success count="1">Deleted 1 AVP Definition(s)</Success>
                <Failure count="0"></Failure>
        </Command>
</Response>
```

Query AVP Definition

The following examples show the request and response that are defined in the XSDs for the QueryAvpDefinition tag.

Request

This request follows the QueryAvpDefinition tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <QueryAvpDefinition>
        <AvpDefinition>
            <Name>AVPSAMPLE:1123</Name>
        </AvpDefinition>
    </QueryAvpDefinition>
</XmlInterfaceRequest>
```

Response

The response to this request follows the **ConfigurationData ➤ AvpDefinition** tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="8.0.0">
    <AvpDefinition>
        <Name>AVPSAMPLE:1123</Name>
        <Description></Description>
        <AvpName>AVPSAMPLE</AvpName>
        <AvpCode>123</AvpCode>
        <VendorId>1123</VendorId>
        <MandatoryFlag>false</MandatoryFlag>
        <ProtectFlag>false</ProtectFlag>
        <MayencryptFlag>false</MayencryptFlag>
        <VendorSpecificFlag>true</VendorSpecificFlag>
        <AvpType>float32</AvpType>
        <RootAvp></RootAvp>
    </AvpDefinition>
</ConfigurationData>
```

Serving GPRS Support Node

A Serving GPRS Support Node (SGSN) may not provide a Gateway GPRS Support Node (GGSN) with accurate or complete mobile country code (MCC) or mobile network code (MNC) information. If not, the GGSN cannot pass this information on to the PCRF (including an MPE device), reducing the PCRF's ability to detect specific roaming scenarios. The MCC/MNC mapping table provides a mechanism for the MPE device to convert an SGSN IP address (a value the GGSN can determine without SGSN input) to the proper MCC/MNC value. You can map multiple serving gateways to each MCC/MNC pair. Once the MCC/MNC values are determined, they can be used in policies to differentiate subscriber treatment based on the specific roaming scenario.

Add SGSN

The following examples show the request and response that are defined in the XSDs for the Add tag.

Request

This request follows the AddSgwMapping tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
```

```
    <AddSgwMapping>
        <SgwMapping>
            <Name>SGSN-1</Name>
            <Description>SGSN-1</Description>
            <MccMnc>12345</MccMnc>
            <SgwIpAddress>
                <IpAddress>10.60.25.33</IpAddress>
            </SgwIpAddress>
        </SgwMapping>
    </AddSgwMapping>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
  <Result>0</Result>
   <Command type="XmlInterfaceResponse">
     <Success count="1">Successfully imported 1 Serving Gateway/MCC-MNC
Mapping(s).</Success>
     <Failure count="0"></Failure>
   </Command>
</Response>
```

Update SGSN

The following examples show the request and response that are defined in the XSDs for the UpdateSgwMapping tag.

Request

This request follows the UpdateSgwMapping tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <UpdateSgwMapping>
        <SgwMapping>
            <Name>SGSN-1</Name>
            <Description>SGSN-1</Description>
            <MccMnc>12345</MccMnc>
            <SgwIpAddress>
                <IpAddress>10.60.25.33</IpAddress>
            </SgwIpAddress>
        </SgwMapping>
    </UpdateSgwMapping>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
   <Result>0</Result>
     <Command type="XmlInterfaceResponse">
        <Success count="1">Successfully updated 1 Serving Gateway/MCC-MNC
Mapping(s).</Success>
        <Failure count="0"></Failure>
```

```
     </Command>
</Response>
```

Delete SGSN

The following examples show the request and response that are defined in the XSDs for the DeleteSwgMapping tag.

Request

This request follows the DeleteSwgMapping tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <DeleteSgwMapping>
        <SgwMapping>
            <Name>SGSN-1</Name>
        </SgwMapping>
</DeleteSgwMapping>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
   <Result>0</Result>
     <Command type="XmlInterfaceResponse">
        <Success count="1">Delete 1 Serving Gateway/MCC-MNC mappings.</Success>
        <Failure count="0"></Failure>
     </Command>
</Response>
```

Query SGSN

The following examples show the request and response that are defined in the XSDs for the QuerySgwmapping tag.

Request

This request follows the QuerySgwmapping tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <QuerySgwMapping>
            <Name>SGSN-1</Name>
    </QuerySgwMapping>
</XmlInterfaceRequest>
```

Response

The response to this request follows the **ConfigurationData ➤ AvpDefinition** tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="8.0.0">
    <SgwMapping>
        <Name>SGSN-1</Name>
        <Description>SGSN-1</Description>
```

```
        <MccMnc>12345</MccMnc>
        <SgwIpAddress>
            <IpAddress>10.60.25.33</IpAddress>
        </SgwIpAddress>
    </SgwMapping>
</ConfigurationData>
```

Chapter 5

Subscriber Interface

Topics:

The Subscriber Interface allows users to manage and query subscriber elements within their system. Subscriber data includes accounts and tiers.

Accounts

An account represents a billable entity that contains the individual users.

Overview

The Subscriber Interface consists of the following requests:

- Accounts — An account represents a billable entity that contains the individual users.
  - Add Account — Allows users to add and update subscriber accounts within their system. If an account already exists with the given identifier (Account ID), then this is considered an update and the new account replaces the existing one. All relationships between that account and other objects in the system (for example, tiers) are maintained.
  - Update Account — Allows users to update accounts in their system. If no object exists with the given identifier (Account ID), then this is a failure condition and the object is not saved.
  - Delete Account — Allows users to delete an account.
  - Query Account — Allows users to query a specific account or list of accounts, returning their configured attributes.

- Tiers — A tier represents a class of service or a logical grouping of subscriber accounts.
  - Add Tier — Allows users to add new tiers to their system. If a tier already exists with the given identifier (Name), then this is considered an update and the new tier replaces the existing one. All relationships between that tier and other objects in the system (for example, accounts) are maintained.
  - Update Tier — Allows users to update tiers within their system. If no object exists with the given identifier (Name), then this is a failure condition and the object is not saved.
  - Delete Tier — Allows users to delete a tier.
  - Query Tier — Allows users to query a specific tier or list of tiers, returning their configured attributes.

Add Account

The following examples show the request and response that are defined in the XSDs for the AddAccount tag.

Request

This request follows the AddAccount tag defined in the XSDs.

The following example creates an account with the following attributes:

- Account ID: 000123
- Association to a network element named: Node1
- One user in this account, with the following attributes:
  - User ID: bill_smith@company.com
- One cable modem in this account, with the following attributes:

- MAC address: 10:10:10:10:10:11
- IP address: 30.0.0.3
- Two CPEs behind this cable modem: 10.0.0.3, 20.0.0.3

- Association to a tier named: Premium

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
<AddAccount>
    <Account>
        <AccountId>000123</AccountId>
        <NetworkElementName>Node1</NetworkElementName>
        <EndUser>
         <EndUserId>john_smith@company.com</EndUserId>
        </EndUser>
        <CableModem>
            <CmMacAddr>10:10:10:10:10:11</CmMacAddr>
            <CmIpAddr>30.0.0.3</CmIpAddr>
            <Cpe>
                <CpeIpAddr>10.0.0.3</CpeIpAddr>
            </Cpe>
            <Cpe>
                <CpeIpAddr>20.0.0.3</CpeIpAddr>
            </Cpe>
        </CableModem>
        <TierRef>
            <Name>Premium</Name>
        </TierRef>
    </Account>
</AddAccount>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse" >
        <Success count="1">Successfully imported 1 accounts.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Update Account

The following examples show the request and response that are defined in the XSDs for the UpdateAccount tag.

Request

This request follows the UpdateAccount tag defined in the XSDs.

The following example updates the account identified by ID 000123 to add a new user with ID jane_doe@company.com.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
<UpdateAccount>
```

```
    <Account>
        <AccountId>000123</AccountId>
        <EndUser operator='add'>
            <EndUserId>jane_doe@company.com</EndUserId>
        </EndUser>
    </Account>
</UpdateAccount>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Successfully updated 1 accounts.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Delete Account

The following examples show the request and response that are defined in the XSDs for the DeleteAccount tag.

Request

This request follows the DeleteAccount tag defined in the XSDs.

The following example deletes two accounts: Account1 and Account2.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
<DeleteAccount>
    <AccountId>Account1</AccountId>
    <AccountId>Account2</AccountId>
</DeleteAccount>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="2">Deleted 2 accounts.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Query Account

The following examples show the request and response that are defined in the XSDs for the QueryAccount tag.

Request

This request follows the QueryAccount tag defined in the XSDs.

The following example queries multiple accounts by account ID.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
    <QueryAccount>
        <AccountId>000123</AccountId>
</QueryAccount>
</XmlInterfaceRequest>
```

Response

The response to this request follows the ConfigurationData -> Account tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="">
    <Account>
        <AccountId>000123</AccountId>
        <NetworkElementName>Node1</NetworkElementName>
        <EndUser>
         <EndUserId>john_smith@company.com</EndUserId>
        </EndUser>
        <CableModem>
            <CmMacAddr>10:10:10:10:10:11</CmMacAddr>
            <CmIpAddr>30.0.0.3</CmIpAddr>
            <UpServerIndex>0</UpServerIndex>
            <UpChannelIndex>0</UpChannelIndex>
            <DownServerIndex>0</DownServerIndex>
            <DownChannelIndex>0</DownChannelIndex>
            <DocsisVersion>0</DocsisVersion>
            <MaxFlows>0</MaxFlows>
            <MaxClassifiers>0</MaxClassifiers>
            <Cpe>
             <CpeMacAddr>00:00:00:00:00:00</CpeMacAddr>
             <CpeIpAddr>10.0.0.3</CpeIpAddr>
            </Cpe>
            <Cpe>
             <CpeMacAddr>00:00:00:00:00:00</CpeMacAddr>
             <CpeIpAddr>20.0.0.3</CpeIpAddr>
            </Cpe>
        </CableModem>
        <TierRef>
            <Name>Premium</Name>
        </TierRef>
    </Account>
</ConfigurationData>
```

Tiers

A tier represents a class of service or a logical grouping of subscriber accounts.

Add Tier

The following examples show the request and response that are defined in the XSDs for the AddTier tag.

Request

This request follows the AddTier tag defined in the XSDs.

The following example creates two tiers:

- Tier1: Upstream bandwidth limit: 1 Mbps
- Tier2: Upstream bandwidth limit: 2 Mbps

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
<AddTier>
    <Tier>
        <Name>Tier1</Name>
        <Description/>
        <ResourceLimit>
            <ResourceType>0</ResourceType>
            <Direction>1</Direction>
            <LimitValue>1000000</LimitValue>
        </ResourceLimit>
    </Tier>
    <Tier>
        <Name>Tier2</Name>
        <Description/>
        <ResourceLimit>
            <ResourceType>0</ResourceType>
            <Direction>1</Direction>
            <LimitValue>2000000</LimitValue>
        </ResourceLimit>
    </Tier>
</AddTier>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="2">Successfully imported 2 tiers.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Update Tier

The following examples show the request and response that are defined in the XSDs for the UpdateTier tag.

Request

This request follows the UpdateTier tag defined in the XSDs.

The following example updates the tier named Tier1 to change the upstream bandwidth limit to 555000 bps.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
    <UpdateTier>
        <Tier>
            <Name>Tier1</Name>
```

```
            <ResourceLimit>
                <ResourceType>0</ResourceType>
                <Direction>1</Direction>
                <LimitValue>555000</LimitValue>
            </ResourceLimit>
        </Tier>
    </UpdateTier>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="1">Successfully updated 1 tier.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Delete Tier

The following examples show the request and response that are defined in the XSDs for the DeleteTier tag.

Request

This request follows the DeleteTier tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
    <DeleteTier>
        <Name>Tier1</Name>
        <Name>Tier2</Name>
    </DeleteTier>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
    <Result>0</Result>
    <Command type="XmlInterfaceResponse">
        <Success count="2">Deleted 2 Tiers.</Success>
        <Failure count="0"></Failure>
    </Command>
</Response>
```

Query Tier

The following examples show the request and response that are defined in the XSDs for the QueryTier tag.

Request

This request follows the QueryTier tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
  <QueryTier>
     <Name>Tier Gold</Name>
     <Name>Tier Bronze</Name>
  </QueryTier>
</XmlInterfaceRequest>
```

Response

The response to this request follows the ConfigurationData -> Tier tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="3.0.0">
    <Tier>
        <Name>Tier Gold</Name>
        <Description/>
        <ResourceLimit>
            <ResourceType>0</ResourceType>
            <Direction>1</Direction>
            <LimitValue>9000000</LimitValue>
        </ResourceLimit>
    </Tier>
    <Tier>
        <Name>Tier Bronze</Name>
        <Description/>
        <ResourceLimit>
            <ResourceType>0</ResourceType>
            <Direction>1</Direction>
            <LimitValue>1000000</LimitValue>
        </ResourceLimit>
    </Tier>
</ConfigurationData>
```

Quota Profiles

A quota set limits on a subscriber's usage; the quota profile defines a specific quota.

Add Quota Profile

The following examples show the request and response that are defined in the XSDs for the AddQuota tag.

Request

This request follows the AddQuota tag defined in the XSDs.

The following example creates two quotas:

- Quota1:
  - quotaType = 0
  - EnableDynamicGrant = true
  - MaxSessionsUsedForDynamicGrant = 20
  - MinGrantSize = 3

- Quota2:
  - quotaType = 1
  - EnableDynamicGrant = false
  - MaxSessionsUsedForDynamicGrant = 10
  - MinGrantSize = 0

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <AddQuota>
        <Quota>
            <Name>quota1</Name>
            <Description></Description>
            <LimitTotalVolume>false</LimitTotalVolume>
            <LimitUpVolume>false</LimitUpVolume>
            <LimitDownVolume>false</LimitDownVolume>
            <TotalVolumeLimit>0</TotalVolumeLimit>
            <UpVolumeLimit>0</UpVolumeLimit>
            <DownVolumeLimit>0</DownVolumeLimit>
            <LimitTime>false</LimitTime>
            <TimeLimit>0</TimeLimit>
            <LimitEvent>false</LimitEvent>
            <EventLimit>0</EventLimit>
            <ReplenishingFrequency>0</ReplenishingFrequency>
            <VolumeThresholdPercentage>0.0</VolumeThresholdPercentage>
            <TimeThresholdPercentage>0.0</TimeThresholdPercentage>
            <EventThresholdPercentage>0.0</EventThresholdPercentage>
            <EnableInterimReporting>false</EnableInterimReporting>
            <InterimReportingInterval>0</InterimReportingInterval>
            <QuotaExhaustionAction>0</QuotaExhaustionAction>
            <RedirectServerType>1</RedirectServerType>
            <QuotaResetIntervalType>1</QuotaResetIntervalType>
            <QuotaResetDayOfWeek>0</QuotaResetDayOfWeek>
            <QuotaResetTimeOfDay></QuotaResetTimeOfDay>
            <QuotaResetTimeVariable>d</QuotaResetTimeVariable>
            <QuotaReportOffsetLimit>0</QuotaReportOffsetLimit>
            <QuotaType>0</QuotaType>
            <MaxLeakageThreshold>1</MaxLeakageThreshold>
            <EnableDynamicGrant>true</EnableDynamicGrant>
            <MaxSessionsUsedForDynamicGrant>20</MaxSessionsUsedForDynamicGrant>
            <MinGrantSize>3</MinGrantSize>
        </Quota>
        <Quota>
            <Name>p1</Name>
            <Description></Description>
            <LimitTotalVolume>false</LimitTotalVolume>
            <LimitUpVolume>false</LimitUpVolume>
            <LimitDownVolume>false</LimitDownVolume>
            <TotalVolumeLimit>0</TotalVolumeLimit>
            <UpVolumeLimit>0</UpVolumeLimit>
            <DownVolumeLimit>0</DownVolumeLimit>
            <LimitTime>false</LimitTime>
            <TimeLimit>0</TimeLimit>
            <LimitEvent>false</LimitEvent>
            <EventLimit>0</EventLimit>
            <ReplenishingFrequency>0</ReplenishingFrequency>
            <VolumeThresholdPercentage>0.0</VolumeThresholdPercentage>
            <TimeThresholdPercentage>0.0</TimeThresholdPercentage>
            <EventThresholdPercentage>0.0</EventThresholdPercentage>
            <EnableInterimReporting>false</EnableInterimReporting>
            <InterimReportingInterval>0</InterimReportingInterval>
            <QuotaExhaustionAction>0</QuotaExhaustionAction>
```

```
            <RedirectServerType>1</RedirectServerType>
            <QuotaResetIntervalType>1</QuotaResetIntervalType>
            <QuotaResetDayOfWeek>0</QuotaResetDayOfWeek>
            <QuotaResetTimeOfDay></QuotaResetTimeOfDay>
            <QuotaResetTimeVariable></QuotaResetTimeVariable>
            <QuotaReportOffsetLimit>0</QuotaReportOffsetLimit>
            <QuotaType>1</QuotaType>
            <MaxLeakageThreshold>1</MaxLeakageThreshold>
            <EnableDynamicGrant>false</EnableDynamicGrant>
            <MaxSessionsUsedForDynamicGrant>10</MaxSessionsUsedForDynamicGrant>
            <MinGrantSize>0</MinGrantSize>
        </Quota>
    </AddQuota>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
   <Result>0</Result>
      <Command type="XmlInterfaceResponse">
         <Success count="2">Successfully imported 2 quota(s).</Success>
         <Failure count="0"></Failure>
      </Command>
</Response>
```

Update Quota Profile

The following examples show the request and response that are defined in the XSDs for the UpdateQuota tag.

Request

This request follows the UpdateQuota tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <UpdateQuota>
        <Quota>
            <Name>quota1</Name>
            <Description></Description>
            <LimitTotalVolume>false</LimitTotalVolume>
            <LimitUpVolume>false</LimitUpVolume>
            <LimitDownVolume>false</LimitDownVolume>
            <TotalVolumeLimit>0</TotalVolumeLimit>
            <UpVolumeLimit>0</UpVolumeLimit>
            <DownVolumeLimit>0</DownVolumeLimit>
            <LimitTime>false</LimitTime>
            <TimeLimit>0</TimeLimit>
            <LimitEvent>false</LimitEvent>
            <EventLimit>0</EventLimit>
            <ReplenishingFrequency>0</ReplenishingFrequency>
            <VolumeThresholdPercentage>0.0</VolumeThresholdPercentage>
            <TimeThresholdPercentage>0.0</TimeThresholdPercentage>
            <EventThresholdPercentage>0.0</EventThresholdPercentage>
            <EnableInterimReporting>false</EnableInterimReporting>
            <InterimReportingInterval>0</InterimReportingInterval>
            <QuotaExhaustionAction>0</QuotaExhaustionAction>
            <RedirectServerType>1</RedirectServerType>
            <QuotaResetIntervalType>1</QuotaResetIntervalType>
```

```
            <QuotaResetDayOfWeek>0</QuotaResetDayOfWeek>
            <QuotaResetTimeOfDay></QuotaResetTimeOfDay>
            <QuotaResetTimeVariable>d</QuotaResetTimeVariable>
            <QuotaReportOffsetLimit>30</QuotaReportOffsetLimit>
            <QuotaType>0</QuotaType>
            <MaxLeakageThreshold>1</MaxLeakageThreshold>
            <EnableDynamicGrant>true</EnableDynamicGrant>
            <MaxSessionsUsedForDynamicGrant>20</MaxSessionsUsedForDynamicGrant>
            <MinGrantSize>3</MinGrantSize>
        </Quota>
        <Quota>
            <Name>p1</Name>
            <Description></Description>
            <LimitTotalVolume>false</LimitTotalVolume>
            <LimitUpVolume>false</LimitUpVolume>
            <LimitDownVolume>false</LimitDownVolume>
            <TotalVolumeLimit>0</TotalVolumeLimit>
            <UpVolumeLimit>0</UpVolumeLimit>
            <DownVolumeLimit>0</DownVolumeLimit>
            <LimitTime>false</LimitTime>
            <TimeLimit>0</TimeLimit>
            <LimitEvent>false</LimitEvent>
            <EventLimit>0</EventLimit>
            <ReplenishingFrequency>0</ReplenishingFrequency>
            <VolumeThresholdPercentage>0.0</VolumeThresholdPercentage>
            <TimeThresholdPercentage>0.0</TimeThresholdPercentage>
            <EventThresholdPercentage>0.0</EventThresholdPercentage>
            <EnableInterimReporting>false</EnableInterimReporting>
            <InterimReportingInterval>0</InterimReportingInterval>
            <QuotaExhaustionAction>0</QuotaExhaustionAction>
            <RedirectServerType>1</RedirectServerType>
            <QuotaResetIntervalType>2</QuotaResetIntervalType>
            <QuotaResetDayOfWeek>0</QuotaResetDayOfWeek>
            <QuotaResetTimeOfDay></QuotaResetTimeOfDay>
            <QuotaResetTimeVariable></QuotaResetTimeVariable>
            <QuotaReportOffsetLimit>0</QuotaReportOffsetLimit>
            <QuotaType>1</QuotaType>
            <MaxLeakageThreshold>1</MaxLeakageThreshold>
            <EnableDynamicGrant>false</EnableDynamicGrant>
            <MaxSessionsUsedForDynamicGrant>10</MaxSessionsUsedForDynamicGrant>
            <MinGrantSize>0</MinGrantSize>
        </Quota>
    </UpdateQuota>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
  <Result>0</Result>
    <Command type="XmlInterfaceResponse">
      <Success count="1">Successfully updated 2 quota.</Success>
      <Failure count="0"></Failure>
    </Command>
</Response>
```

Delete Quota Profile

The following examples show the request and response that are defined in the XSDs for the DeleteQuota tag.

Request

This request follows the DeleteQuota tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" ?>
  <XmlInterfaceRequest>
    <DeleteQuota>
      <Name> quota1</Name>
    </DeleteQuota>
</XmlInterfaceRequest>
```

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
<Result>0</Result>
   <Command type="XmlInterfaceResponse">
      <Success count="1>Deleted 1 quota</Success>
      <Failure count="0"></Failure>
   </Command>
</Response>
```

Query Quota Profile

The following examples show the request and response that are defined in the XSDs for the QueryQuota tag.

Request

This request follows the QueryQuota tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" ?>
  <XmlInterfaceRequest>
    <QueryQuota>
       <Name> quota1</Name>
    </QueryQuota>
</XmlInterfaceRequest>
```

Response

The response to this request follows the **ConfigurationData ➤ Quota** tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="8.0.0">
    <Quota>
        <Name>quota1</Name>
        <Description></Description>
        <LimitTotalVolume>false</LimitTotalVolume>
        <LimitUpVolume>false</LimitUpVolume>
        <LimitDownVolume>false</LimitDownVolume>
        <TotalVolumeLimit>0</TotalVolumeLimit>
        <UpVolumeLimit>0</UpVolumeLimit>
        <DownVolumeLimit>0</DownVolumeLimit>
        <LimitTime>false</LimitTime>
        <TimeLimit>0</TimeLimit>
        <LimitEvent>false</LimitEvent>
        <EventLimit>0</EventLimit>
        <ReplenishingFrequency>0</ReplenishingFrequency>
        <VolumeThresholdPercentage>0.0</VolumeThresholdPercentage>
```

```
        <TimeThresholdPercentage>0.0</TimeThresholdPercentage>
        <EventThresholdPercentage>0.0</EventThresholdPercentage>
        <EnableInterimReporting>false</EnableInterimReporting>
        <InterimReportingInterval>0</InterimReportingInterval>
        <QuotaExhaustionAction>0</QuotaExhaustionAction>
        <RedirectServerType>2</RedirectServerType>
        <QuotaResetIntervalType>1</QuotaResetIntervalType>
        <QuotaResetDayOfMonth>0</QuotaResetDayOfMonth>
        <QuotaResetDayOfWeek>0</QuotaResetDayOfWeek>
        <QuotaResetTimeOfDay></QuotaResetTimeOfDay>
        <QuotaResetTimeVariable>d</QuotaResetTimeVariable>
        <QuotaReportOffsetLimit>0</QuotaReportOffsetLimit>
        <QuotaType>0</QuotaType>
        <MaxLeakageThreshold>0</MaxLeakageThreshold>
        <EnableDynamicGrant>false</EnableDynamicGrant>
        <MaxSessionsUsedForDynamicGrant>20</MaxSessionsUsedForDynamicGrant>
        <MinGrantSize>0</MinGrantSize>
    </Quota>
</ConfigurationData>
```

Setting SCTP Settings

When enabled, this feature allows you to set diameter connections using SCTP.

To enable or disable processing of SCTP procedures, or to change configuration attributes:

1. From the **Policy Server** section of the navigation pane, select **Global Configuration Settings**.
   The content tree displays a list of global configuration settings.
2. From the content tree, select **SCTP Settings**.
   The SCTP Settings page opens in the work area set to *True*.
3. On the SCTP Settings page, click **Modify**.
   The SCTP Settings page opens.
4. Enter a value for the configuration attribute:
   a) **SCTP Enabled** — Check mark to enable this feature. It will automatically begin listening for incoming SCTP connections.

      Note: If datasource is not supported by SCTP, then the default is unchecked (disabled) for a TCP connection.

5. When you finish, click **Save** (or **Cancel** to discard your changes).
   The SCTP Settings page closes.

The SCTP Settings attribute is configured.

Chapter 6

Operational Measurements Interface Overview

Topics:

The Operational Measurements (OM) XML interface is used to retrieve operational counters from the system. This chapter describes the interface and how it should be used.

OM Statistics Scheduled Task

The OM interface requires that the "OM Statistics" scheduled task be running on the Configuration Management Platform (CMP). This task performs the function of collecting the operational counters from the MPEs in the network and recording them in the CMP database; the data is then available to be queried via the OM XML interface. This task can be configured by the user to poll at intervals between 5 minutes and 24 hours, with a default value of 15 minutes, and keeps the data available for query for 1 to 30 days, with the default value being 7 days. The recommended settings for this task varies with each deployment, dependant on the volume of data being collected.

When OM requests are made, the data for the response is taken from the information that has been collected by this task. Without the data that is collected through this scheduled task, there would be no data available for OM queries.

Most values returned as part of the response are presented (by default) as deltas, representing the positive change between the start time and end time. In order to calculate a delta there must be a minimum of two recorded values available, so the OM Statistics task must have been run at least twice in order to provide any data through the OM XML interface.

OM Statistics Requests

OM statistic requests consist of the following:

- Attributes and Child Tag
- Recorded Timestamp and Request Time Range

Attributes and Child Tags

The following attributes can be used with any of the OM Statistics requests as part of the "QueryOMStats" tag:

- DeltaCount — The values for statistics which are cumulative in nature (for example, monotonically increasing counters) are returned by default as a delta value. For these cumulative statistics, the delta value returned represents the positive change in that value since the last recorded time period. By setting the DeltaCount attribute to "false" the user can request that all statistics be returned as absolute values instead, that is the total values since the beginning of time.

  Some statistics are always returned as absolute values. These non-cumulative statistics have values that can increase or decrease over time, for example Upstream and Downstream Bandwidth or active Session Counts. These statistics are explicitly called out in this document as always returning "absolute" values.

  The following is an example of the DeltaCount attribute:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats DeltaCount="false">
    <StartTime>2006-10-12T11:15:00Z</StartTime>
    <EndTime>2006-10-12T11:25:00Z</EndTime>
```

```
    <NetworkElementStats/>
</QueryOmStats>
```

- AggregateTimeSamples — The AggregateTimeSamples attribute allows the user to request that all sample buckets in the response be aggregated into a single sample bucket. This allows users to request data for a time period and see a summary of all recorded data for that specific time period, rather than all the individual samples that make up the requested time period.

  The following is an example of the AggregateTimeSamples attribute:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats AggregateTimeSamples="true">
    <StartTime>2006-10-12T11:15:00Z</StartTime>
    <EndTime>2006-10-12T11:25:00Z</EndTime>
    <NetworkElementStats/>
</QueryOmStats>
```

The following child tags are available for OM XML Commands:

- StartTime/EndTime — These tags define the time range over which you want statistical data returned. All statistic data recorded by the OM Statistics task over that requested time range is returned. Also, the EndTime tag is an optional tag. If no EndTime tag is specified, then the time range extends from the StartTime and continues until the current time. This provides a means to capture the most up-to-date data.

  The following is an example of the StartTime/EndTime child tags:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:35:00Z</EndTime>
    <PolicyServerStats/>
    <IsComplete>true</IsComplete>
</QueryOmStats>
```

- PolicyServer — This optional tag, specified as a child tag of some individual statistic groups, allows the user to request statistics specific to an individual or a set of MPEs. The default behavior displays statistics for all MPEs.

  The following is an example of the PolicyServer child tag:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:35:00Z</EndTime>
    <PolicyServerStats>
        <PolicyServer>Atlanta105</PolicyServer>
      <IsComplete>true</IsComplete>
    </PolicyServerStats>
</QueryOmStats>
```

Recorded Timestamp and Request Time Range

The OM Statistics task runs on a scheduled interval. The statistics, however, are time stamped as they are received from the MPEs to increase accuracy. For example, while the OM Statistics task may start at 12:00:00, the statistics recorded for MPE1 may not be time stamped until 12:00:01, and MPE2 at 12:00:02, and so on.

This detail impacts the way the captured data should be requested through the XML interface so as to ensure all the desired data is returned as part of the response. To ensure correct results, if the intent is to retrieve data for a full 24 hour period, then the start and end times should extend beyond the intended 24 hour period to ensure that any delayed results returned for an MPE are included in the response.

For example, to retrieve statistics recorded for the entire day of March 10, the following request time range is recommended (note that the EndTime is 15 minutes past midnight):

```
<StartTime>2008-03-10T00:00:00Z</StartTime>
<EndTime>2008-03-11T00:15:00Z</EndTime>
```

OM Statistics Response Format

The XSD defines the schema for responses to OM XML requests. Results are returned in "Sample" buckets representing a range of time. If Persistent Interval Statistics are enabled (Stats.Interval.Enabled), this range is determined by the **Stats Collection Period**, found under **Global Configuration Settings ➤ Stats Settings** on the CMP, but also includes any "Manual" runs of the task initiated by the user.

If Stats.Interval.Enabled is set to false (meaning that Persistent Interval Statistics are not enabled), then the CMP collects absolute counter values only; this range is determined by the interval settings of the OM Statistics scheduled task and includes any "Manual" runs of the task initiated by the user.

Responses do not return data sets that exceed a maximum size. In this case, a user is directed to reduce the scope of their query to accommodate for the size limitation of the XML response.

Interval Statistics

In the interval-statistics mode, all numeric data generated by MPEs and MRAs are reset at regular intervals controlled by the Stats Collection Period. When in Interval mode, a reset occurs on the hour and then every 5, 10, 15, 20, 30 or 60 minutes afterwards depending on the value of the Stats Collection Period. See the *CMP Wireless User Guide* for details on setting interval statistics.

When in Interval Mode, the XML request and response will look similar to the example below.

Request:

```
<QueryOmStatsSetting/>
```

Response:

```
<OmStatsSetting>
   <ResetConfiguration>Interval</ResetConfiguration>
   <CollectionInterval>15</CollectionInterval>
</OmStatsSetting>
```

Absolute versus Delta Values

For all cumulative statistics (for example, monotonically increasing counters), the returned values are presented as deltas, by default. These deltas represent the positive change in value between the start and end times. This behavior can optionally be changed to return absolute values for all statistics (see *OM Statistics Requests*).

Non-cumulative statistics, those statistics whose value can either increase or decrease between intervals, are returned as absolute values. This prevents negative value responses as a result of a delta calculation. For example, in "NetworkElementStats" the Upstream Bandwidth, Downstream Bandwidth, and "active" Session Counts are returned as absolute values. These absolute statistics are explicitly called out in this document as always returning "absolute" values. All absolute data points are taken as the value at the End Time of the reported Sample.

Empty Data Set

Requests that do not result in any statistics contain the following message:

```
Not enough statistical data available to fulfill request.
```

If you receive this message, check the request parameters or select a different time range, as the message indicates that for the parameters submitted as part of the request there is not enough recorded data to respond.

This response can occur for a number of reasons. If you see the following in the response:

```
<IsComplete>false</IsComplete>
```

then the MPE or MRA has detected errors when collecting the stats, usually because a complete interval was not available.

The OM Statistics Task must have been run at least two times within the specified time range (StartTime and EndTime). The task itself runs on a schedule but the data is recorded dynamically as it is received from the MPEs so this should be taken into account when requesting a particular range of time. Tekelec recommends that users start with a larger time window for testing and narrow that down, adjusting the scheduled task as needed, until the desired data is returned. This message could also occur if recorded data is unavailable for a specific parameter, such as a certain NetworkElement or MPE name.

Timezones and Start/End Times

OM responses include "Sample" buckets, each with a specified start and end time. These values are returned in UTC (Coordinated Universal Time) format, which is the international time standard. The CMP is unaware of the timezone of the originating request and is therefore unable to return a "localized" timezone to the user.

The XSDs define request/responses with the following:

```
<!-- Time Range -->
<xsd:element name="StartTime" type="xsd:dateTime" minOccurs="1" maxOccurs="1"/>
<xsd:element name="EndTime" type="xsd:dateTime" minOccurs="1" maxOccurs="1"/>
```

The type is defined as "`xsd:dateTime`". This is a UTC format. This data type describes instances identified by the combination of a date and a time.

Its lexical space is the extended format:

```
[-]CCYY-MM-DDThh:mm:ss[Z|(+|-)hh:mm]
```

The time zone may be specified as Z (UTC) or (+|-)hh:mm. Time zones that are not specified are considered localized to the MPE Manager.

The following are examples of valid values for xsd:dateTime:

Example 1:

```
2007-03-26T21:32:52Z - UTC format. This is the default response format for
all timestamps. It is also the recommend format for user requests.
```

Example 2:

```
2007-03-26T21:32:52+02:00 - Localized time with 2 hour offset to UTC time.
```

Example 3:

```
2007-03-26T19:32:52 - Localized time. This will be interpreted as localized
to the MPE Manager.
```

Example 4:

```
2007-03-26T19:32:52+00:00 - Localized time with no offset.
```

Note: All date and time numbers are two digits. For example, 3:00am, April 5th is written as "2008-04-05T03:00:00" and not "2008-4-5T3:00:00".

A request to retrieve statistics can be made either in UTC or as a time that is localized to the CMP. Tekelec recommends that users make their requests in UTC format to remain consistent with the UTC output. UTC is also recommended, as end users may not be aware they are making requests from a different timezone then the CMP and therefore may not be accounting for that difference when analyzing results. If the MPEs, CMP, and end user are all in different timezones then the user would be attempting to correlate values and draw conclusions from the reported statistics across all of those timezones.

Counter Reset and Failover

MPEs and MRAs write historical data to a distributed comcol database and save it for 24 hours.

Note: CMP stores cluster-level interval stats to its own database. Data is available for external OSS system to retrieve after the end of the next interval period.

There are two cases where statistics can be reset:

- The user can click the "Reset All Counters" button on the Policy Server "Reports" page of the CMP Graphical User Interface (GUI). This button is typically used to reset counters as a baseline from which to run tests (refer to *CMP User Guide* for detailed description of this page.).
- The user can enable the Interval mode; when configured, numeric values are reset at regular intervals controlled by the Stats Collection Period. When in Interval mode, a reset will occur on the hour and then every 5, 10, 15, 20, 30 or 60 minutes afterwards depending on the value of the Stats Collection Period. (When interval periods are changed, the database is purged.) See the *CMP User Guide* for details.

In either case, all memory counters are reset to 0. This means that the delta values calculated for the time period during which the reset occurred are not accurate.

For example, if at the StartTime the SuccessSessionCount value was 100, and a reset occurred resetting the count to 0, then the next reported delta would be -100. If the SuccessSessionCount started at 100, and during the sample time period, five successful sessions completed, then a reset occurred after which 10 more sessions completed successfully, then the resulting delta for that time period would be -90. This example shows that some information is lost; the response will indicate this with the

`IsComplete` flag set to `false`. In either case, the value could not be seen as an accurate representation of activity for that time and should be ignored. For this reason, negative delta values are always returned as "0".

The CMP will get incomplete interval stats from MPE/MRA in the following conditions:

- The active MPE/MRA blade switches over one or more times within an interval period.
- MPE/MRA has initially started up.
- Under high-load conditions, the MPE/MRA cannot store data to the database at the end of an interval cycle (within a margin of a few seconds).

Comparisons Between the CMP GUI and OM Statistics

Relationships between the OM XML responses and the CMP Reports GUI can be drawn and used for comparison.

The OM XML interface persists statistic values over time. This tool is intended for historical analysis of statistics and can be used to track usage. The interface allows users to request data over a user-defined time range and returns data for that period. The default behavior for this interface is to return data as delta values. The delta is calculated as the positive change in value between the start and end times returned in "Sample" buckets. Certain non-cumulative statistics are always reported as absolutes and those individual statistics are explicitly documented. This behavior can also be changed to return absolute values for all statistics; even those which are cumulative in nature (see *OM Statistics Requests*).

The "Reports" page in the Policy Server section of the CMP GUI displays statistics in real time (Refer to the *CMP User's Guide* for details). The intent of this page is for monitoring current statistics. The page dynamically updates the displayed statistics every 10 seconds. These statistics are categorized by protocol. The default behavior of this page displays statistics as absolute values. There is a "Show Deltas" button at the top of the page; this button can be used to switch the display to calculated delta values. The delta value shown is the difference between the current value and the last refresh of the page (approximately 10 seconds).

The top section on any statistics page, contained within the CMP GUI, usually correlates with the "Message Processing Stats" of the OM XML interface. The individual statistics for each Network Element can be found in the section at the bottom of the reporting page. These statistics correlate with the "Network Element Stats" of the OM XML interface.

Labels for specific statistics may differ between the CMP GUI and the OM XML interface responses. The specific types of statistics displayed may also differ by protocol. While the CMP GUI can use labels and statistics specific to a certain protocol, the OM XML interface requests must use generic language across protocols in order to remain consistent with published XSD definitions.

For deployments that collect statistics across multiple protocols, for example PCMM and DQoS, the CMP GUI displays statistics for each protocol individually. However, in the XML Interface, the statistics for available protocols are summarized in the response. For example, session count statistics for PCMM and DQoS would be added together to display with each network element. Statistics for individual protocols are not be displayed separately.

Additional details for the GUI's Reports/Statistics section of the CMP GUI can be found in the *CMP User's Guide*.

Chapter 7

Operational Measurement Requests

Topics:

This chapter defines the various Operational Measurements (OM) groups and the individual OM statistics that comprise them. For several of the more generic statistics, the instrumentation on the Multimedia Policy Engine (MPE) and the Multi-Protocol Routing Agent (MRA) may differ by protocol, and therefore specific statistics may increment differently across those protocols.

Operational Measurements Requests Overview

The OM interface consists of the following requests, grouped by category:

- OSSI XML interface statistics
  - Topology Update Statistics — retrieves statistics on all updates made through the Topology Interface.
  - Subscriber Update Statistics — retrieves statistics on all updates made through the Subscriber Interface.
  - Policy Server Statistics — retrieves statistics on each policy server in the system, including the associated subscribers and network elements.
- Protocol-specific statistics
  - Message Processing Statistics — retrieves statistics on session events.
  - Network Element Statistics — retrieves statistics on session events for specific network elements and interfaces.
  - Reserve Commit Statistics — retrieves statistics on reserve and commit requests specific to the SPC DQoS protocol.
  - Gate Statistics — retrieves statistics on gate set messages, gate status messages, and gate delete messages specific to PCMM, DQoS and SPC DQoS protocols.
  - RADIUS-S Statistics — retrieves statistics on accounting and change-of-authorization messages specific to the RADIUS-S protocol.
  - Connected Network Element Statistics — retrieves statistics, reporting the number of connected network elements on a per-MPE basis. Connected network elements are those that have a connection (for example, COPS) established to the MPE. This statistic is specific to wireline customers.
  - Diameter Statistics — retrieves statistics on Diameter Application Functions, Charging Functions, and Policy Charging Enforcement Functions.
  - Timeout Statistics — retrieves timeout statistics not received in a predefined amount of time, that are tracked per network element, MPE, and MRA.
  - Latency Statistics — retrieves latency statistics information for incoming and outgoing messages tracked per network element, MPE, and MRA.
  - Event Trigger Statistics — retrieves related event trigger statistics used for tracking activity.
  - Session Cleanup Statistics — retrieves the number of session cleanup statistics where sessions are aged-out due to inactivity. The reauthorization is submitted and if it is unresponsive or an error occurs indicating that the session is inactive, then it is removed from the system.
  - Traffic Profile Statistics — retrieves traffic profile statistics which counters the number of times a PCRF attempts to install, remove, or fails.
  - Quota Profile Statistics — retrieves quota profile statistics to counter the number of times the configured threshold reaches a maxed out limit.

Event Trigger Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterEventTriggerStats tag.

Request

This request follows the QueryOmStats -> DiameterEventTriggerStats tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2001-12-31T12:00:00</StartTime>
    <EndTime>2001-12-31T12:15:00</EndTime>
    <DiameterEventTriggerStats>
      <PolicyServer>PolicyServer</PolicyServer>
    </DiameterEventTriggerStats>
</QueryOmStats>
```

The response to this request follows the Statistics -> DiameterEventTriggerStats tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterEventTriggerStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <SGSN_CHANGE>1</SGSN_CHANGE>
      <QOS_CHANGE>1</QOS_CHANGE>
      <RAT_CHANGE>1</RAT_CHANGE>
      <TFT_CHANGE>1</TFT_CHANGE>
      <PLMN_CHANGE>1</PLMN_CHANGE>
      <LOSS_OF_BEARER>1</LOSS_OF_BEARER>
      <RECOVERY_OF_BEARER>1</RECOVERY_OF_BEARER>
      <IP_CAN_CHANGE>1</IP_CAN_CHANGE>
      <GW_PCEF_MALFUNCTION>1</GW_PCEF_MALFUNCTION>
      <RESOURCES_LIMITATION>1</RESOURCES_LIMITATION>
      <MAX_NR_BEARERS_REACHED>1</MAX_NR_BEARERS_REACHED>
      <QOS_CHANGE_EXCEEDING_AUTHORIZATION>1</QOS_CHANGE_EXCEEDING_AUTHORIZATION>
      <RAI_CHANGE>1</RAI_CHANGE>
      <USER_LOCATION_CHANGE>1</USER_LOCATION_CHANGE>
      <OUT_OF_CREDIT>1</OUT_OF_CREDIT>
      <REALLOCATION_OF_CREDIT>1</REALLOCATION_OF_CREDIT>
      <REVALIDATION_TIMEOUT>1</REVALIDATION_TIMEOUT>
      <UE_IP_ADDRESS_ALLOCATE>1</UE_IP_ADDRESS_ALLOCATE>
      <UE_IP_ADDRESS_RELEASE>1</UE_IP_ADDRESS_RELEASE>
      <DEFAULT_EPS_BEARER_QOS_CHANGE>1</DEFAULT_EPS_BEARER_QOS_CHANGE>
      <AN_GW_CHANGE>1</AN_GW_CHANGE>
      <SUCCESSFUL_RESOURCE_ALLOCATION>1</SUCCESSFUL_RESOURCE_ALLOCATION>
      <UE_TIME_ZONE_CHANGE>1</UE_TIME_ZONE_CHANGE>
      <USAGE_REPORT>1</USAGE_REPORT>
      <USAGE_THRESHOLD_REACHED>1</USAGE_THRESHOLD_REACHED>
      <SERVICE_FLOW_DETECTION>1</SERVICE_FLOW_DETECTION>
    </SAMPLE>
```

```
    <IsComplete>true</IsComplete>
  </DiameterEventTriggerStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterConnectionEventTriggerStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
  <StartTime>2001-12-31T12:00:00</StartTime>
  <EndTime>2001-12-31T12:15:00</EndTime>
  <DiameterConnectionEventTriggerStats>
    <Name>NetworkElementName</Name>
    <NeId>NeId</NeId>
    <PolicyServer>PolicyServer</PolicyServer>
  </DiameterConnectionEventTriggerStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterConnectionEventTriggerStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterConnectionEventTriggerStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <Name>NetworkElementName</Name>
      <NeId>NeId</NeId>
      <SGSN_CHANGE>1</SGSN_CHANGE>
      <QOS_CHANGE>1</QOS_CHANGE>
      <RAT_CHANGE>1</RAT_CHANGE>
      <TFT_CHANGE>1</TFT_CHANGE>
      <PLMN_CHANGE>1</PLMN_CHANGE>
      <LOSS_OF_BEARER>1</LOSS_OF_BEARER>
      <RECOVERY_OF_BEARER>1</RECOVERY_OF_BEARER>
      <IP_CAN_CHANGE>1</IP_CAN_CHANGE>
      <GW_PCEF_MALFUNCTION>1</GW_PCEF_MALFUNCTION>
      <RESOURCES_LIMITATION>1</RESOURCES_LIMITATION>
      <MAX_NR_BEARERS_REACHED>1</MAX_NR_BEARERS_REACHED>
      <QOS_CHANGE_EXCEEDING_AUTHORIZATION>1</QOS_CHANGE_EXCEEDING_AUTHORIZATION>
      <RAI_CHANGE>1</RAI_CHANGE>
      <USER_LOCATION_CHANGE>1</USER_LOCATION_CHANGE>
      <OUT_OF_CREDIT>1</OUT_OF_CREDIT>
      <REALLOCATION_OF_CREDIT>1</REALLOCATION_OF_CREDIT>
      <REVALIDATION_TIMEOUT>1</REVALIDATION_TIMEOUT>
      <UE_IP_ADDRESS_ALLOCATE>1</UE_IP_ADDRESS_ALLOCATE>
      <UE_IP_ADDRESS_RELEASE>1</UE_IP_ADDRESS_RELEASE>
      <DEFAULT_EPS_BEARER_QOS_CHANGE>1</DEFAULT_EPS_BEARER_QOS_CHANGE>
      <AN_GW_CHANGE>1</AN_GW_CHANGE>
      <SUCCESSFUL_RESOURCE_ALLOCATION>1</SUCCESSFUL_RESOURCE_ALLOCATION>
      <UE_TIME_ZONE_CHANGE>1</UE_TIME_ZONE_CHANGE>
      <USAGE_REPORT>1</USAGE_REPORT>
```

```
      <USAGE_THRESHOLD_REACHED>1</USAGE_THRESHOLD_REACHED>
      <SERVICE_FLOW_DETECTION>1</SERVICE_FLOW_DETECTION>
    </SAMPLE>
  </DiameterConnectionEventTriggerStats>
</Statistics>
```

Note: Only non-zero event trigger counts are included in the response.

Individual statistics are defined as follows for the PCEF and BBERF protocols:

- SGSN_CHANGE (0)
- QOS_CHANGE (1)
- RAT_CHANGE (2)
- TFT_CHANGE (3)
- PLMN_CHANGE (4)
- LOSS_OF_BEARER (5)
- RECOVERY_OF_BEARER (6)
- IP_CAN_CHANGE (7)
- GW_PCEF_MALFUNCTION (8)
- RESOURCES_LIMITATION (9)
- MAX_NR_BEARERS_REACHED (10)
- QOS_CHANGE_EXCEEDING_AUTHORIZATION (11)
- RAI_CHANGE (12)
- USER_LOCATION_CHANGE (13)
- OUT_OF_CREDIT (14)
- REALLOCATION_OF_CREDIT (15)
- REVALIDATION_TIMEOUT (16)
- UE_IP_ADDRESS_ALLOCATE (17)
- UE_IP_ADDRESS_RELEASE (18)
- DEFAULT_EPS_BEARER_QOS_CHANGE (20)
- AN_GW_CHANGE (21)

Topology Update Statistics

The following examples show the request and response that are defined in the XSDs for the TopologyUpdateStats tag.

Request

This request follows the **QueryOmStats ➤ TopologyUpdateStats** tag defined in the XSDs. For example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:40:00Z</EndTime>
    <TopologyUpdateStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ TopologyUpdateStats** tag defined in the XSDs.

- TopologyUpdateCount — The number of topology changes made in the Configuration Management Platform (CMP) that resulted in changes to an MPE. Topology data includes Network Elements, Paths, Interfaces, and Links. Changes to this data can occur through either the CMP Graphical User Interface (GUI) or the XML interface. The counter increments for each MPE that updates as a result of the change. For example, when a user updates a Network Element, the topology update counter increments for each MPE that the Network Element is associated with.

  Batch changes are treated as a single update to an MPE. An OSSI XML interface update may be made across multiple elements but all of those changes are pushed at one time to the MPE. This is treated as a single topology change with regard to the counter.
- TopologyUpdateFailCount — The number of topology changes made in the CMP that fail to update an associated MPE. For example, if an MPE is offline while a Network Element is changed, the fail counter will increment once for that failed MPE update.

The following is an example of the TopologyUpdateStats tag:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
    <TopologyUpdateStats>
        <Sample>
       <StartTime>2006-10-26T14:30:22Z</StartTime>
       <EndTime>2006-10-26T14:35:11Z</EndTime>
       <TopologyUpdateCount>942</TopologyUpdateCount>
       <TopologyUpdateFailCount>6</TopologyUpdateFailCount>
        </Sample>
    </TopologyUpdateStats>
</Statistics>
```

Subscriber Update Statistics

The following examples show the request and response that are defined in the XSDs for the SubscriberUpdateStats tag.

Request

This request follows the **QueryOmStats ➤ SubscriberUpdateStats** tag defined in the XSDs. For example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:40:00Z</EndTime>
    <SubscriberUpdateStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ SubscriberUpdateStats** tag defined in the XSDs.

- SubscriberUpdateCount — The number of subscriber changes made in the CMP that resulted in changes to an MPE. Subscriber data includes accounts and tiers. Changes to this data can occur through either the CMP GUI or the XML Interface. The counter is incremented for each MPE that updates as a result of the change. For example, when a user updates an account, the subscriber update counter increments for each MPE that the account is associated with.

Batch changes are treated as a single update to an MPE. An OSSI XML interface update may be made across multiple accounts but all of those changes are pushed at one time to the MPE. This is treated as a single subscriber change with regard to the counter.

- SubscriberUpdateFailCount — The number of subscriber changes made in the CMP that fail to update an associated MPE. For example, if an MPE is offline while an account is changed, the fail counter will increment once for that failed MPE update.

The following is an example of the SubscriberUpdateStats tag:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
    <SubscriberUpdateStats>
        <Sample>
       <StartTime>2006-10-26T14:30:22Z</StartTime>
       <EndTime>2006-10-26T14:35:11Z</EndTime>
       <SubscriberUpdateCount>942</SubscriberUpdateCount>
       <SubscriberUpdateFailCount>6</SubscriberUpdateFailCount>
        </Sample>
    </SubscriberUpdateStats>
</Statistics>
```

Policy Server Statistics

The following examples show the request and response that are defined in the XSDs for the PolicyServerStats tag.

Request

This request follows the **QueryOmStats ➤ PolicyServerStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:35:00Z</EndTime>
    <PolicyServerStats>
        <PolicyServer>Atlanta105</PolicyServer>
    </PolicyServerStats>
</QueryOmStats>
```

The following is an example of a request for all policy servers in the system:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:40:00Z</EndTime>
    <PolicyServerStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ PolicyServerStats** tag defined in the XSDs.

- TotalNetworkElementCount — The total (absolute) number of network elements associated with that MPE. The absolute value is the value taken at the EndTime of the reported sample.

- TotalSubscriberCount — The total (absolute) number of accounts associated with that MPE.

The following is an example of the PolicyServerStats tag:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
    <PolicyServerStats>
        <Sample>
       <StartTime>2006-10-26T14:30:22Z</StartTime>
       <EndTime>2006-10-26T14:35:11Z</EndTime>
       <PolicyServer>Atlanta105</PolicyServer>
       <IsComplete>true</IsComplete>
       <TotalNetworkElementCount>52</TotalNetworkElementCount>
       <TotalSubscriberCount>1000000</TotalSubscriberCount>
        </Sample>
    </PolicyServerStats>
</Statistics>
```

Message Processing Statistics

The following examples show the request and response that are defined in the XSDs for the MessageProcessingStats tag.

Request

This request follows the **QueryOmStats ➤ MessageProcessingStats** tag defined in the XSDs.

The following is an example of a request for all policy servers in the system:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:40:00Z</EndTime>
    <MessageProcessingStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ MessageProcessingStats** tag defined in the XSDs.

- SessionCount — Session requests received.
- SessionSuccessCount — Session requests successfully created.
- SessionFailCount — Session request failures. For example, this could be due to a policy denying a request.
- SessionProtocolFailCount — Number of session failures due to an invalid message or parameter. This count is incremented whenever the MPE determines that an incoming message from the AM has an invalid message and has to be dropped by the MPE.
- SessionPolicyFailCount — Number of session requests that trigger a policy. This count is maintained in the MPE, one per policy. When the condition of a policy triggers, the count for that policy is incremented. The value displayed is a total trigger count. That is, the sum of this value for all the policies. For example, if the definition of a policy is defined as, “when the device usage is greater than 80% of capacity, reject message”. When the MPE executes this policy in response to a request, if the policy triggers (i.e. if the MPE determines that the device usage is > 80%), the trigger count for that policy is incremented.

The following is an example of the MessageProcessingStats tag:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
    <MessageProcessingStats>
      <Sample>
       <StartTime>2006-10-26T14:30:22Z</StartTime>
       <EndTime>2006-10-26T14:35:11Z</EndTime>
       <PolicyServer>Atlanta105</PolicyServer>
       <IsComplete>true</IsComplete>
       <SessionCount>100</SessionCount>
       <SessionSuccessCount>97</SessionSuccessCount>
       <SessionFailCount>3</SessionFailCount>
       <SessionProtocolFailCount>1</SessionProtocolFailCount>
       <SessionPolicyFailCount>2</SessionPolicyFailCount>
      </Sample>
    </MessageProcessingStats>
</Statistics>
```

Network Element Statistics

The following examples show the request and response that are defined in the XSDs for the NetworkElementStats tag.

Request

This request follows the **QueryOmStats ➤ NetworkElementStats** tag defined in the XSDs.

The following is an example of a request for a single network element using the Name parameter:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:35:00Z</EndTime>
    <NetworkElementStats>
        <Name>Router1</Name>
    </NetworkElementStats>
</QueryOmStats>
```

The following is an example of a request for multiple network elements using the Name(s) and NeId(s) parameters. This example returns statistics for three different network elements:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2006-10-12T11:15:00Z</StartTime>
    <EndTime>2006-10-12T11:25:00Z</EndTime>
    <NetworkElementStats>
        <Name>Router1</Name>
        <Name>Router2</Name>
        <Name>Router3</Name>
    </NetworkElementStats>
</QueryOmStats>
```

The following is an example of a request for all network elements in the system. This example returns statistics for each network element:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<QueryOmStats>
    <StartTime>2006-10-12T11:15:00Z</StartTime>
    <EndTime>2006-10-12T11:25:00Z</EndTime>
    <NetworkElementStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ NetworkElementStats** tag defined in the XSDs.

Both Name and NeId are returned along with the statistics and actual recorded times for those statistics. Sample groups are ordered by policy server, network element, available interfaces, and then by time.

This OM Group, by default, contains a combination of delta and absolute values. Individual statistics returned as absolutes are described as follows:

- Name — Unique name identifying the Network Element for the following statistics.
  - NeId — Optional identifier field for the Network Element.
  - InterfaceName — Identifying field for the Network Element's Interface.
  - SessionCount — Current active sessions for that Network Element or Interface. This is a non-cumulative value and is displayed as an absolute.
  - SessionSuccessCount — Successful sessions.
  - SessionFailCount — Session failures.
  - AbnormalDisconnectCount — Number of network elements that have disconnected from the MPE abnormally. For example, due to a break in the network.
  - Capacity — The currently defined maximum capacity for this Network Element or Interface. This is a static absolute value defined in the CMP for that object.
  - BandwidthUpstream — This is the current reserved upstream bandwidth allocated for this Network Element or Interface. This is a non-cumulative value displayed as an absolute.
  - BandwidthDownstream — This is the current reserved downstream bandwidth allocated for this Network Element or Interface. This is a non-cumulative value displayed as an absolute.

The following is an example XML response to a multiple network element request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
    <NetworkElementStats>
        <Sample>
            <StartTime>2006-10-12T11:18:30Z</StartTime>
            <EndTime>2006-10-12T11:19:20Z</EndTime>
            <PolicyServer>Atlanta105</PolicyServer>
            <IsComplete>true</IsComplete>
            <Name>Router1</Name>
            <NeId>12345</NeId>
            <SessionCount>3</SessionCount>
            <SessionSuccessCount>3</SessionSuccessCount>
            <SessionFailCount>0</SessionFailCount>
            <AbnormalDisconnectCount>0</AbnormalDisconnectCount>
            <Capacity>50000000</Capacity>
            <BandwidthUpstream>3000000</BandwidthUpstream>
            <BandwidthDownstream>100000</BandwidthDownstream>
            <Interface>
                <InterfaceName>If1</InterfaceName>
                <SessionCount>3</SessionCount>
                <SessionSuccessCount>3</SessionSuccessCount>
                <SessionFailCount>0</SessionFailCount>
                <AbnormalDisconnectCount>0</AbnormalDisconnectCount>
                <Capacity>50000000</Capacity>
```

```
            <BandwidthUpstream>3000000</BandwidthUpstream>
            <BandwidthDownstream>100000</BandwidthDownstream>
        </Interface>
    </Sample>
    <Sample>
        <StartTime>2006-10-12T11:19:20Z</StartTime>
        <EndTime>2006-10-12T11:20:10Z</EndTime>
        <PolicyServer>Atlanta105</PolicyServer>
        <IsComplete>true</IsComplete>
        <Name>Router1</Name>
        <NeId>12345</NeId>
        <SessionCount>0</SessionCount>
        <SessionSuccessCount>0</SessionSuccessCount>
        <SessionFailCount>0</SessionFailCount>
        <AbnormalDisconnectCount>0</AbnormalDisconnectCount>
        <Capacity>50000000</Capacity>
        <BandwidthUpstream>3000000</BandwidthUpstream>
        <BandwidthDownstream>100000</BandwidthDownstream>
        <Interface>
            <InterfaceName>If1</InterfaceName>
            <SessionCount>0</SessionCount>
            <SessionSuccessCount>0</SessionSuccessCount>
            <SessionFailCount>0</SessionFailCount>
            <AbnormalDisconnectCount>0</AbnormalDisconnectCount>
            <Capacity>50000000</Capacity>
            <BandwidthUpstream>3000000</BandwidthUpstream>
            <BandwidthDownstream>100000</BandwidthDownstream>
        </Interface>
    </Sample>
    <Sample>
        <StartTime>2006-10-12T11:20:10Z</StartTime>
        <EndTime>2006-10-12T11:21:00Z</EndTime>
        <PolicyServer>Atlanta105</PolicyServer>
        <IsComplete>true</IsComplete>
        <Name>Router1</Name>
        <NeId>12345</NeId>
        <SessionCount>9</SessionCount>
        <SessionSuccessCount>9</SessionSuccessCount>
        <SessionFailCount>0</SessionFailCount>
        <AbnormalDisconnectCount>0</AbnormalDisconnectCount>
        <Capacity>50000000</Capacity>
        <BandwidthUpstream>3000000</BandwidthUpstream>
        <BandwidthDownstream>100000</BandwidthDownstream>
        <Interface>
            <InterfaceName>If1</InterfaceName>
            <SessionCount>9</SessionCount>
            <SessionSuccessCount>9</SessionSuccessCount>
            <SessionFailCount>0</SessionFailCount>
            <AbnormalDisconnectCount>0</AbnormalDisconnectCount>
            <Capacity>50000000</Capacity>
            <BandwidthUpstream>3000000</BandwidthUpstream>
            <BandwidthDownstream>100000</BandwidthDownstream>
        </Interface>
    </Sample>
    <Sample>
        <StartTime>2006-10-12T11:18:30Z</StartTime>
        <EndTime>2006-10-12T11:19:20Z</EndTime>
        <PolicyServer>Atlanta105</PolicyServer>
        <IsComplete>true</IsComplete>
        <Name>Router 2</Name>
        <NeId>12341</NeId>
        <SessionCount>11</SessionCount>
        <SessionSuccessCount>11</SessionSuccessCount>
        <SessionFailCount>0</SessionFailCount>
        <AbnormalDisconnectCount>0</AbnormalDisconnectCount>
```

```
            <Capacity>50000000</Capacity>
            <BandwidthUpstream>3000000</BandwidthUpstream>
            <BandwidthDownstream>100000</BandwidthDownstream>
            <Interface>
                <InterfaceName>If2</InterfaceName>
                <SessionCount>11</SessionCount>
                <SessionSuccessCount>11</SessionSuccessCount>
                <SessionFailCount>0</SessionFailCount>
                <AbnormalDisconnectCount>0</AbnormalDisconnectCount>
                <Capacity>50000000</Capacity>
                <BandwidthUpstream>3000000</BandwidthUpstream>
                <BandwidthDownstream>100000</BandwidthDownstream>
            </Interface>
        </Sample>
        <Sample>
            <StartTime>2006-10-12T11:19:20Z</StartTime>
            <EndTime>2006-10-12T11:20:10Z</EndTime>
            <PolicyServer>Atlanta105</PolicyServer>
            <IsComplete>true</IsComplete>
            <Name>Router 2</Name>
            <NeId>12341</NeId>
            <SessionCount>2</SessionCount>
            <SessionSuccessCount>2</SessionSuccessCount>
            <SessionFailCount>0</SessionFailCount>
            <AbnormalDisconnectCount>0</AbnormalDisconnectCount>
            <Capacity>50000000</Capacity>
            <BandwidthUpstream>3000000</BandwidthUpstream>
            <BandwidthDownstream>100000</BandwidthDownstream>
            <Interface>
                <InterfaceName>If2</InterfaceName>
                <SessionCount>2</SessionCount>
                <SessionSuccessCount>2</SessionSuccessCount>
                <SessionFailCount>0</SessionFailCount>
                <AbnormalDisconnectCount>0</AbnormalDisconnectCount>
                <Capacity>50000000</Capacity>
                <BandwidthUpstream>3000000</BandwidthUpstream>
                <BandwidthDownstream>100000</BandwidthDownstream>
            </Interface>
        </Sample>
        <Sample>
            <StartTime>2006-10-12T11:20:10Z</StartTime>
            <EndTime>2006-10-12T11:21:00Z</EndTime>
            <PolicyServer>Atlanta105</PolicyServer>
            <IsComplete>true</IsComplete>
            <Name>Router 2</Name>
            <NeId>12341</NeId>
            <SessionCount>9</SessionCount>
            <SessionSuccessCount>9</SessionSuccessCount>
            <SessionFailCount>0</SessionFailCount>
            <AbnormalDisconnectCount>0</AbnormalDisconnectCount>
            <Capacity>50000000</Capacity>
            <BandwidthUpstream>3000000</BandwidthUpstream>
            <BandwidthDownstream>100000</BandwidthDownstream>
            <Interface>
                <InterfaceName>If2</InterfaceName>
                <SessionCount>9</SessionCount>
                <SessionSuccessCount>9</SessionSuccessCount>
                <SessionFailCount>0</SessionFailCount>
                <AbnormalDisconnectCount>0</AbnormalDisconnectCount>
                <Capacity>50000000</Capacity>
                <BandwidthUpstream>3000000</BandwidthUpstream>
                <BandwidthDownstream>100000</BandwidthDownstream>
            </Interface>
        </Sample>
```

```
    </NetworkElementStats>
</Statistics>
```

Reserve Commit Statistics

The following examples show the request and response that are defined in the XSDs for the ReserveCommitStats tag.

Request

This request follows the **QueryOmStats ➤ ReserveCommitStats** tag defined in the XSDs.

The following is an example of a request for all policy servers in the system:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2007-10-26T14:30:00Z</StartTime>
    <EndTime>2007-10-26T14:40:00Z</EndTime>
    <ReserveCommitStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ ReserveCommitStats** tag defined in the XSDs.

- ReserveCount — Number of GateSet Reserve messages received.
- ReserveSuccessCount — Number of GateSet Reserve messages acknowledged.
- ReserveFailCount — Number of GateSet Reserve messages failed.
- ReserveErrorCount — Number of GateSet Reserve messages that were errors.
- CommitCount — Number of GateSet Commit messages received.
- CommitSuccessCount — Number of GateSet Commit messages acknowledged.
- CommitFailCount — Number of GateSet Commit messages failed.
- CommitErrorCount — Number of GateSet Commit messages that were errors.
- ReserveCommitCount — Number of GateSet Reserve+Commit messages received.
- ReserveCommitSuccessCount — Number of GateSet Reserve+Commit messages acknowledged.
- ReserveCommitFailCount — Number of GateSet Reserve+Commit messages failed.
- ReserveCommitErrorCount — Number of GateSet Reserve+Commit messages that were errors.

The following is an example XML response to a ReserveCommitStats tag request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
    <ReserveCommitStats>
      <Sample>
       <StartTime>2007-10-26T14:30:22Z</StartTime>
       <EndTime>2007-10-26T14:35:11Z</EndTime>
       <PolicyServer>Atlanta105</PolicyServer>
       <IsComplete>true</IsComplete>
       <ReserveCount>100</ReserveCount>
       <ReserveSuccessCount>99</ReserveSuccessCount>
       <ReserveFailCount>1</ReserveFailCount>
       <ReserveErrorCount>0</ReserveErrorCount>
       <CommitCount>99</CommitCount>
       <CommitSuccessCount>99</CommitSuccessCount>
       <CommitFailCount>0</CommitFailCount>
```

```
       <CommitErrorCount>0</CommitErrorCount>
       <ReserveCommitCount>99</ReserveCommitCount>
       <ReserveCommitSuccessCount>99</ReserveCommitSuccessCount>
       <ReserveCommitFailCount>0</ReserveCommitFailCount>
       <ReserveCommitErrorCount>0</ReserveCommitErrorCount>
      </Sample>
    </ReserveCommitStats>
</Statistics>
```

Gate Statistics

The following examples show the request and response that are defined in the XSDs for the GateStats tag.

Request

This request follows the **QueryOmStats ➤ GateStats** tag defined in the XSDs.

The following is an example of a request for all policy servers in the system:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2007-10-26T14:30:00Z</StartTime>
    <EndTime>2007-10-26T14:40:00Z</EndTime>
    <GateStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ GateStats** tag defined in the XSDs.

- GateSetCount — Number of Gate Set messages processed.
- GateSetSuccessCount — Number of Gate Set Success messages processed.
- GateSetErrorCount — Number of Gate Set Error messages processed.
- GateStatusCount — Number of Gate Status messages processed.
- GateStatusSuccessCount — Number of Gate Status Success messages processed.
- GateStatusErrorCount — Number of Gate Status Error messages processed.
- GateDeleteCount — Number of Gate Delete messages processed.
- GateDeleteSuccessCount — Number of Gate Delete Success messages processed.
- GateDeleteErrorCount — Number of Gate Delete Error messages processed.

The following is an example XML response to a GateStats tag request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
    <GateStats>
      <Sample>
       <StartTime>2007-10-26T14:30:22Z</StartTime>
       <EndTime>2007-10-26T14:35:11Z</EndTime>
       <PolicyServer>Atlanta105</PolicyServer>
       <IsComplete>true</IsComplete>
       <GateSetCount>100</GateSetCount>
       <GateSetSuccessCount>100</GateSetSuccessCount>
       <GateSetErrorCount>0</GateSetErrorCount>
       <GateStatusCount>100</GateStatusCount>
       <GateStatusSuccessCount>100</GateStatusSuccessCount>
```

```
       <GateStatusErrorCount>0</GateStatusErrorCount>
       <GateDeleteCount>100</GateDeleteCount>
       <GateDeleteSuccessAckCount>100</GateDeleteSuccessCount>
       <GateDeleteErrorCount>0</GateDeleteErrorCount>
      </Sample>
    </GateStats>
</Statistics>
```

Latency Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterAfLatencyStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterAfLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2001-12-31T12:00:00</StartTime>
    <EndTime>2001-12-31T12:15:00</EndTime>
    <DiameterAfLatencyStats>
      <PolicyServer>PolicyServer</PolicyServer>
    </DiameterAfLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterAfLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterAfLatencyStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
   </Sample>
```

```
 </DiameterAfLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterAfPeerLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2001-12-31T12:00:00</StartTime>
    <EndTime>2001-12-31T12:15:00</EndTime>
    <DiameterAfPeerLatencyStats>
       <Name>NetworkElementName</Name>
       <NeId>NeId</NeId>
    </DiameterAfPeerLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterAfPeerfLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterAfPeerfLatencyStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <Name>NetworkElementName</Name>
      <NeId>NeId</NeId>
      <NetworkElementType>NetworkElementType</NetworkElementType>
      <NetworkElementSubType>NetworkElementSubType</NetworkElementSubType>
      <ConnectAddress>10.60.4.56</ConnectAddress>
      <ConnectPort>3868</ConnectPort>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterAfPeerfLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterPcefLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterPcefLatencyStats>
        <PolicyServer>PolicyServer</PolicyServer>
```

```
      </DiameterPcefLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterPcefLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterPcefLatencyStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterPcefLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterPcefPeerLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterPcefPeerLatencyStats>
        <Name>NetworkElementName</Name>
        <NeId>NeId</NeId>
      </DiameterPcefPeerLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterPcefPeerLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterPcefPeerLatencyStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
```

```
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <Name>NetworkElementName</Name>
      <NeId>NeId</NeId>
      <NetworkElementType>NetworkElementType</NetworkElementType>
      <NetworkElementSubType>NetworkElementSubType</NetworkElementSubType>
      <ConnectAddress>10.60.4.56</ConnectAddress>
      <ConnectPort>3868</ConnectPort>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterPcefPeerLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterBberfLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterBberfLatencyStats>
        <PolicyServer>PolicyServer</PolicyServer>
      </DiameterBberfLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterBberfLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterBberfLatencyStats >
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
```

```
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterBberfLatencyStats >
</Statistics>
```

The response to this request follows the **Statistics ➤ DiameterBberfLatencyStats** tag defined in the XSDs.

The following is an example of a response for an MPE:

```
<Statistics>
  <DiameterBberfLatencyStats>
   <Sample>
      <StartTime>2012-06-07T01:00:00Z</StartTime>
      <EndTime>2012-06-07T01:15:00Z</EndTime>
      <PolicyServer>kli-63-mpe</PolicyServer>
      <IsComplete>true</IsComplete>
      <CurrentConnectionsCount>1</CurrentConnectionsCount>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>

      <TransactionTime_In_0_20_Count>0</TransactionTime_In_0_20_Count>
      <TransactionTime_In_20_40_Count>0</TransactionTime_In_20_40_Count>
      <TransactionTime_In_40_60_Count>0</TransactionTime_In_40_60_Count>
      <TransactionTime_In_60_80_Count>0</TransactionTime_In_60_80_Count>
      <TransactionTime_In_80_100_Count>0</TransactionTime_In_80_100_Count>
      <TransactionTime_In_100_120_Count>0</TransactionTime_In_100_120_Count>
      <TransactionTime_In_120_140_Count>0</TransactionTime_In_120_140_Count>
      <TransactionTime_In_140_160_Count>0</TransactionTime_In_140_160_Count>
      <TransactionTime_In_160_180_Count>0</TransactionTime_In_160_180_Count>
      <TransactionTime_In_180_200_Count>0</TransactionTime_In_180_200_Count>
      <TransactionTime_In_gt_200_Count>0</TransactionTime_In_gt_200_Count>
      <TransactionTime_Out_0_20_Count>0</TransactionTime_Out_0_20_Count>
      <TransactionTime_Out_20_40_Count>0</TransactionTime_Out_20_40_Count>
      <TransactionTime_Out_40_60_Count>0</TransactionTime_Out_40_60_Count>
      <TransactionTime_Out_60_80_Count>0</TransactionTime_Out_60_80_Count>
      <TransactionTime_Out_80_100_Count>0</TransactionTime_Out_80_100_Count>
      <TransactionTime_Out_100_120_Count>0</TransactionTime_Out_100_120_Count>
      <TransactionTime_Out_120_140_Count>0</TransactionTime_Out_120_140_Count>
      <TransactionTime_Out_140_160_Count>0</TransactionTime_Out_140_160_Count>
      <TransactionTime_Out_160_180_Count>0</TransactionTime_Out_160_180_Count>
      <TransactionTime_Out_180_200_Count>0</TransactionTime_Out_180_200_Count>
      <TransactionTime_Out_gt_200_Count>0</TransactionTime_Out_gt_200_Count>
    </Sample>
  </DiameterBberfLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterBberfPeerLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterBberfPeerLatencyStats>
        <Name>NetworkElementName</Name>
        <NeId>NeId</NeId>
      </DiameterBberfPeerLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterBberfPeerLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterBberfPeerLatencyStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <Name>NetworkElementName</Name>
      <NeId>NeId</NeId>
      <NetworkElementType>NetworkElementType</NetworkElementType>
      <NetworkElementSubType>NetworkElementSubType</NetworkElementSubType>
      <ConnectAddress>10.60.4.56</ConnectAddress>
      <ConnectPort>3868</ConnectPort>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterBberfPeerLatencyStats>
</Statistics>
```

The response to this request follows the **Statistics ➤ DiameterBberfPeerLatencyStats** tag defined in the XSDs.

The following is an example of a response for an MPE:

```
<Statistics>
  <DiameterBberfPeerLatencyStats>
      <Sample>
      <StartTime>2012-06-07T01:00:00Z</StartTime>
      <EndTime>2012-06-07T01:15:00Z</EndTime>
      <PolicyServer>kli-63-mpe</PolicyServer>
```

```
      <IsComplete>true</IsComplete>
      <Name>mra66</Name>
      <NeId />
      <NetworkElementType />
      <NetworkElementSubType />
      <ConnectAddress>10.60.25.66</ConnectAddress>
      <ConnectPort>48526</ConnectPort>
      <ConnectType>TCP</ConnectType>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>

      <TransactionTime_In_0_20_Count>0</TransactionTime_In_0_20_Count>
      <TransactionTime_In_20_40_Count>0</TransactionTime_In_20_40_Count>
      <TransactionTime_In_40_60_Count>0</TransactionTime_In_40_60_Count>
      <TransactionTime_In_60_80_Count>0</TransactionTime_In_60_80_Count>
      <TransactionTime_In_80_100_Count>0</TransactionTime_In_80_100_Count>
      <TransactionTime_In_100_120_Count>0</TransactionTime_In_100_120_Count>
      <TransactionTime_In_120_140_Count>0</TransactionTime_In_120_140_Count>
      <TransactionTime_In_140_160_Count>0</TransactionTime_In_140_160_Count>
      <TransactionTime_In_160_180_Count>0</TransactionTime_In_160_180_Count>
      <TransactionTime_In_180_200_Count>0</TransactionTime_In_180_200_Count>
      <TransactionTime_In_gt_200_Count>0</TransactionTime_In_gt_200_Count>
      <TransactionTime_Out_0_20_Count>0</TransactionTime_Out_0_20_Count>
      <TransactionTime_Out_20_40_Count>0</TransactionTime_Out_20_40_Count>
      <TransactionTime_Out_40_60_Count>0</TransactionTime_Out_40_60_Count>
      <TransactionTime_Out_60_80_Count>0</TransactionTime_Out_60_80_Count>
      <TransactionTime_Out_80_100_Count>0</TransactionTime_Out_80_100_Count>
      <TransactionTime_Out_100_120_Count>0</TransactionTime_Out_100_120_Count>
      <TransactionTime_Out_120_140_Count>0</TransactionTime_Out_120_140_Count>
      <TransactionTime_Out_140_160_Count>0</TransactionTime_Out_140_160_Count>
      <TransactionTime_Out_160_180_Count>0</TransactionTime_Out_160_180_Count>
      <TransactionTime_Out_180_200_Count>0</TransactionTime_Out_180_200_Count>
      <TransactionTime_Out_gt_200_Count>0</TransactionTime_Out_gt_200_Count>
      </Sample>
   </DiameterBberfPeerLatencyStats>
 </Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterTdfLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
  <StartTime>2001-12-31T12:00:00</StartTime>
  <EndTime>2001-12-31T12:15:00</EndTime>
  <DiameterTdfLatencyStats>
    <PolicyServer>PolicyServer</PolicyServer>
  </DiameterTdfLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterTdfLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<Statistics>
  <DiameterTdfLatencyStats>
    <Sample>
```

```
      <StartTime>2012-06-05T04:45:00Z</StartTime>
      <EndTime>2012-06-05T05:00:00Z</EndTime>
      <PolicyServer>kli-63-mpe</PolicyServer>
      <IsComplete>true</IsComplete>
      <CurrentConnectionsCount>1</CurrentConnectionsCount>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_In_0_20_Count>0</TransactionTime_In_0_20_Count>
      <TransactionTime_In_20_40_Count>0</TransactionTime_In_20_40_Count>
      <TransactionTime_In_40_60_Count>0</TransactionTime_In_40_60_Count>
      <TransactionTime_In_60_80_Count>0</TransactionTime_In_60_80_Count>
      <TransactionTime_In_80_100_Count>0</TransactionTime_In_80_100_Count>
      <TransactionTime_In_100_120_Count>0</TransactionTime_In_100_120_Count>
      <TransactionTime_In_120_140_Count>0</TransactionTime_In_120_140_Count>
      <TransactionTime_In_140_160_Count>0</TransactionTime_In_140_160_Count>
      <TransactionTime_In_160_180_Count>0</TransactionTime_In_160_180_Count>
      <TransactionTime_In_180_200_Count>0</TransactionTime_In_180_200_Count>
      <TransactionTime_In_gt_200_Count>0</TransactionTime_In_gt_200_Count>
      <TransactionTime_Out_0_20_Count>0</TransactionTime_Out_0_20_Count>
      <TransactionTime_Out_20_40_Count>0</TransactionTime_Out_20_40_Count>
      <TransactionTime_Out_40_60_Count>0</TransactionTime_Out_40_60_Count>
      <TransactionTime_Out_60_80_Count>0</TransactionTime_Out_60_80_Count>
      <TransactionTime_Out_80_100_Count>0</TransactionTime_Out_80_100_Count>
      <TransactionTime_Out_100_120_Count>0</TransactionTime_Out_100_120_Count>
      <TransactionTime_Out_120_140_Count>0</TransactionTime_Out_120_140_Count>
      <TransactionTime_Out_140_160_Count>0</TransactionTime_Out_140_160_Count>
      <TransactionTime_Out_160_180_Count>0</TransactionTime_Out_160_180_Count>
      <TransactionTime_Out_180_200_Count>0</TransactionTime_Out_180_200_Count>
      <TransactionTime_Out_gt_200_Count>0</TransactionTime_Out_gt_200_Count>
    </Sample>
  </DiameterTdfLatencyStats>
</Statistics>
```

The response to this request follows the **Statistics ➤ DiameterTdfLatencyStats** tag defined in the XSDs.

The following is an example of a response for an MPE:

```
<Statistics>
  <DiameterTdfLatencyStats>
      <Sample>
      <StartTime>2012-06-05T04:45:00Z</StartTime>
      <EndTime>2012-06-05T05:00:00Z</EndTime>
      <PolicyServer>kli-63-mpe</PolicyServer>
      <IsComplete>true</IsComplete>
      <CurrentConnectionsCount>1</CurrentConnectionsCount>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>

      <TransactionTime_In_0_20_Count>0</TransactionTime_In_0_20_Count>
      <TransactionTime_In_20_40_Count>0</TransactionTime_In_20_40_Count>
      <TransactionTime_In_40_60_Count>0</TransactionTime_In_40_60_Count>
      <TransactionTime_In_60_80_Count>0</TransactionTime_In_60_80_Count>
      <TransactionTime_In_80_100_Count>0</TransactionTime_In_80_100_Count>
      <TransactionTime_In_100_120_Count>0</TransactionTime_In_100_120_Count>
      <TransactionTime_In_120_140_Count>0</TransactionTime_In_120_140_Count>
      <TransactionTime_In_140_160_Count>0</TransactionTime_In_140_160_Count>
      <TransactionTime_In_160_180_Count>0</TransactionTime_In_160_180_Count>
      <TransactionTime_In_180_200_Count>0</TransactionTime_In_180_200_Count>
      <TransactionTime_In_gt_200_Count>0</TransactionTime_In_gt_200_Count>
```

```
      <TransactionTime_Out_0_20_Count>0</TransactionTime_Out_0_20_Count>
      <TransactionTime_Out_20_40_Count>0</TransactionTime_Out_20_40_Count>
      <TransactionTime_Out_40_60_Count>0</TransactionTime_Out_40_60_Count>
      <TransactionTime_Out_60_80_Count>0</TransactionTime_Out_60_80_Count>
      <TransactionTime_Out_80_100_Count>0</TransactionTime_Out_80_100_Count>
      <TransactionTime_Out_100_120_Count>0</TransactionTime_Out_100_120_Count>
      <TransactionTime_Out_120_140_Count>0</TransactionTime_Out_120_140_Count>
      <TransactionTime_Out_140_160_Count>0</TransactionTime_Out_140_160_Count>
      <TransactionTime_Out_160_180_Count>0</TransactionTime_Out_160_180_Count>
      <TransactionTime_Out_180_200_Count>0</TransactionTime_Out_180_200_Count>
      <TransactionTime_Out_gt_200_Count>0</TransactionTime_Out_gt_200_Count>
      </Sample>
    </DiameterTdfLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterTdfPeerLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
 <StartTime>2001-12-31T12:00:00</StartTime>
 <EndTime>2001-12-31T12:15:00</EndTime>
 <DiameterBberfPeerLatencyStats>
   <Name>NetworkElementName</Name>
   <NeId>NeId</NeId>
 </DiameterBberfPeerLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterTdfPeerLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<Statistics>
  <DiameterTdfPeerLatencyStats>
    <Sample>
      <StartTime>2012-05-28T09:15:00Z</StartTime>
      <EndTime>2012-05-28T09:30:00Z</EndTime>
      <PolicyServer>kli-63-mpe</PolicyServer>
      <IsComplete>true</IsComplete>
      <Name>kli-66-mra2</Name>
      <NeId/>
      <NetworkElementType/>
      <NetworkElementSubType/>
      <ConnectAddress>10.60.25.67:57455,10.60.25.68:57455</ConnectAddress>
      <ConnectPort>0</ConnectPort>
      <ConnectType>SCTP</ConnectType>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_In_0_20_Count>0</TransactionTime_In_0_20_Count>
      <TransactionTime_In_20_40_Count>0</TransactionTime_In_20_40_Count>
      <TransactionTime_In_40_60_Count>0</TransactionTime_In_40_60_Count>
      <TransactionTime_In_60_80_Count>0</TransactionTime_In_60_80_Count>
      <TransactionTime_In_80_100_Count>0</TransactionTime_In_80_100_Count>
      <TransactionTime_In_100_120_Count>0</TransactionTime_In_100_120_Count>
      <TransactionTime_In_120_140_Count>0</TransactionTime_In_120_140_Count>
      <TransactionTime_In_140_160_Count>0</TransactionTime_In_140_160_Count>
```

```
      <TransactionTime_In_160_180_Count>0</TransactionTime_In_160_180_Count>
      <TransactionTime_In_180_200_Count>0</TransactionTime_In_180_200_Count>
      <TransactionTime_In_gt_200_Count>0</TransactionTime_In_gt_200_Count>
      <TransactionTime_Out_0_20_Count>0</TransactionTime_Out_0_20_Count>
      <TransactionTime_Out_20_40_Count>0</TransactionTime_Out_20_40_Count>
      <TransactionTime_Out_40_60_Count>0</TransactionTime_Out_40_60_Count>
      <TransactionTime_Out_60_80_Count>0</TransactionTime_Out_60_80_Count>
      <TransactionTime_Out_80_100_Count>0</TransactionTime_Out_80_100_Count>
      <TransactionTime_Out_100_120_Count>0</TransactionTime_Out_100_120_Count>
      <TransactionTime_Out_120_140_Count>0</TransactionTime_Out_120_140_Count>
      <TransactionTime_Out_140_160_Count>0</TransactionTime_Out_140_160_Count>
      <TransactionTime_Out_160_180_Count>0</TransactionTime_Out_160_180_Count>
      <TransactionTime_Out_180_200_Count>0</TransactionTime_Out_180_200_Count>
      <TransactionTime_Out_gt_200_Count>0</TransactionTime_Out_gt_200_Count>
    </Sample>
  </DiameterTdfPeerLatencyStats>
</Statistics>
```

The response to this request follows the **Statistics ➤ DiameterTdfPeerLatencyStats** tag defined in the XSDs.

The following is an example of a response for an MPE:

```
<Statistics>
  <DiameterTdfPeerLatencyStats>
     <Sample>
     <StartTime>2012-05-28T09:15:00Z</StartTime>
     <EndTime>2012-05-28T09:30:00Z</EndTime>
     <PolicyServer>kli-63-mpe</PolicyServer>
     <IsComplete>true</IsComplete>
     <Name>kli-66-mra2</Name>
     <NeId />
     <NetworkElementType />
     <NetworkElementSubType />
     <ConnectAddress>10.60.25.67:57455,10.60.25.68:57455</ConnectAddress>
     <ConnectPort>0</ConnectPort>
     <ConnectType>SCTP</ConnectType>
     <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
     <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
     <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
     <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
     <TransactionTime_In_0_20_Count>0</TransactionTime_In_0_20_Count>
     <TransactionTime_In_20_40_Count>0</TransactionTime_In_20_40_Count>
     <TransactionTime_In_40_60_Count>0</TransactionTime_In_40_60_Count>
     <TransactionTime_In_60_80_Count>0</TransactionTime_In_60_80_Count>
     <TransactionTime_In_80_100_Count>0</TransactionTime_In_80_100_Count>
     <TransactionTime_In_100_120_Count>0</TransactionTime_In_100_120_Count>
     <TransactionTime_In_120_140_Count>0</TransactionTime_In_120_140_Count>
     <TransactionTime_In_140_160_Count>0</TransactionTime_In_140_160_Count>
     <TransactionTime_In_160_180_Count>0</TransactionTime_In_160_180_Count>
     <TransactionTime_In_180_200_Count>0</TransactionTime_In_180_200_Count>
     <TransactionTime_In_gt_200_Count>0</TransactionTime_In_gt_200_Count>
     <TransactionTime_Out_0_20_Count>0</TransactionTime_Out_0_20_Count>
     <TransactionTime_Out_20_40_Count>0</TransactionTime_Out_20_40_Count>
     <TransactionTime_Out_40_60_Count>0</TransactionTime_Out_40_60_Count>
     <TransactionTime_Out_60_80_Count>0</TransactionTime_Out_60_80_Count>
     <TransactionTime_Out_80_100_Count>0</TransactionTime_Out_80_100_Count>
     <TransactionTime_Out_100_120_Count>0</TransactionTime_Out_100_120_Count>
     <TransactionTime_Out_120_140_Count>0</TransactionTime_Out_120_140_Count>
     <TransactionTime_Out_140_160_Count>0</TransactionTime_Out_140_160_Count>
     <TransactionTime_Out_160_180_Count>0</TransactionTime_Out_160_180_Count>
     <TransactionTime_Out_180_200_Count>0</TransactionTime_Out_180_200_Count>
     <TransactionTime_Out_gt_200_Count>0</TransactionTime_Out_gt_200_Count>
     </Sample>
```

```
   </DiameterTdfPeerLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterShLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterShLatencyStats>
        <PolicyServer>PolicyServer</PolicyServer>
      </DiameterShLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterShLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterShLatencyStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterShLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterShPeerLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterShPeerLatencyStats >
        <Name>NetworkElementName</Name>
```

```
        <NeId>NeId</NeId>
      </DiameterShPeerLatencyStats >
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterShPeerLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterShPeerLatencyStats >
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <Name>NetworkElementName</Name>
      <NeId>NeId</NeId>
      <NetworkElementType>NetworkElementType</NetworkElementType>
      <NetworkElementSubType>NetworkElementSubType</NetworkElementSubType>
      <ConnectAddress>10.60.4.56</ConnectAddress>
      <ConnectPort>3868</ConnectPort>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterShPeerLatencyStats >
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterDrmaLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterDrmaLatencyStats>
        <PolicyServer>PolicyServer</PolicyServer>
        <NeId>NeId</NeId>
      </DiameterDrmaLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterDrmaLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterDrmaLatencyStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterDrmaLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterDrmaPeerLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterDrmaPeerLatencyStats>
        <Name>NetworkElementName</Name>
        <NeId>NeId</NeId>
      </DiameterDrmaPeerLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterDrmaPeerLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterDrmaPeerLatencyStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <Name>NetworkElementName</Name>
      <NeId>NeId</NeId>
      <NetworkElementType>NetworkElementType</NetworkElementType>
      <NetworkElementSubType>NetworkElementSubType</NetworkElementSubType>
```

```
      <ConnectAddress>10.60.4.56</ConnectAddress>
      <ConnectPort>3868</ConnectPort>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterDrmaPeerLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterMraAfLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterMraAfLatencyStats>
        <MRA>MRA</MRA>
      </DiameterMraAfLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraAfLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterMraAfLatencyStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <MRA>MRA</MRA>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
```

```
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterMraAfLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterMraAfPeerLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterMraAfPeerLatencyStats>
        <Name>NetworkElementName</Name>
        <NeId>NeId</NeId>
      </DiameterMraAfPeerLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraAfPeerLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterMraAfPeerLatencyStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <MRA>MRA</MRA>
      <Name>NetworkElementName</Name>
      <NeId>NeId</NeId>
      <NetworkElementType>NetworkElementType</NetworkElementType>
      <NetworkElementSubType>NetworkElementSubType</NetworkElementSubType>
      <ConnectAddress>10.60.4.56</ConnectAddress>
      <ConnectPort>3868</ConnectPort>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
   </Sample>
```

```
 </DiameterMraAfPeerLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterMraPcefLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterMraPcefLatencyStats>
        <MRA>MRA</MRA>
      </DiameterMraPcefLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraPcefLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterMraPcefLatencyStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
      <MRA>MRA</MRA>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterMraPcefLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterMraPcefPeerLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
```

```
      <DiameterMraPcefPeerLatencyStats>
        <Name>NetworkElementName</Name>
        <NeId>NeId</NeId>
     </DiameterMraPcefPeerLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraPcefPeerLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterMraPcefPeerLatencyStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <MRA>MRA</MRA>
      <Name>NetworkElementName</Name>
      <NeId>NeId</NeId>
      <NetworkElementType>NetworkElementType</NetworkElementType>
      <NetworkElementSubType>NetworkElementSubType</NetworkElementSubType>
      <ConnectAddress>10.60.4.56</ConnectAddress>
      <ConnectPort>3868</ConnectPort>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterMraPcefPeerLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterMraBberfLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterMraBberfLatencyStats>
        <MRA>MRA</MRA>
      </DiameterMraBberfLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraBberfLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterMraBberfLatencyStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <MRA>MRA</MRA>
      <IsComplete>true</IsComplete>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterMraBberfLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterMraBberfPeerLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterMraBberfPeerLatencyStats >
        <Name>NetworkElementName</Name>
        <NeId>NeId</NeId>
      </DiameterMraBberfPeerLatencyStats >
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraBberfPeerLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterMraBberfPeerLatencyStats >
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <MRA>MRA</MRA>
      <IsComplete>true</IsComplete>
      <Name>NetworkElementName</Name>
      <NeId>NeId</NeId>
```

```
      <NetworkElementType>NetworkElementType</NetworkElementType>
      <NetworkElementSubType>NetworkElementSubType</NetworkElementSubType>
      <ConnectAddress>10.60.4.56</ConnectAddress>
      <ConnectPort>3868</ConnectPort>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterMraBberfPeerLatencyStats >
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterMraTdfLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
 <StartTime>2001-12-31T12:00:00</StartTime>
 <EndTime>2001-12-31T12:15:00</EndTime>
 <DiameterMraTdfLatencyStats>
    <MRA>MRA</MRA>
 </DiameterMraTdfLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraTdfLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<Statistics>
  <DiameterMraTdfLatencyStats>
    <Sample>
      <StartTime>2012-05-29T01:00:00Z</StartTime>
      <EndTime>2012-05-29T01:15:00Z</EndTime>
      <MRA>kli-66-mra</MRA>
      <IsComplete>true</IsComplete>
      <CurrentConnectionsCount>1</CurrentConnectionsCount>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_In_0_20_Count>0</TransactionTime_In_0_20_Count>
      <TransactionTime_In_20_40_Count>0</TransactionTime_In_20_40_Count>
      <TransactionTime_In_40_60_Count>0</TransactionTime_In_40_60_Count>
      <TransactionTime_In_60_80_Count>0</TransactionTime_In_60_80_Count>
      <TransactionTime_In_80_100_Count>0</TransactionTime_In_80_100_Count>
      <TransactionTime_In_100_120_Count>0</TransactionTime_In_100_120_Count>
      <TransactionTime_In_120_140_Count>0</TransactionTime_In_120_140_Count>
```

```
      <TransactionTime_In_140_160_Count>0</TransactionTime_In_140_160_Count>
      <TransactionTime_In_160_180_Count>0</TransactionTime_In_160_180_Count>
      <TransactionTime_In_180_200_Count>0</TransactionTime_In_180_200_Count>
      <TransactionTime_In_gt_200_Count>0</TransactionTime_In_gt_200_Count>
      <TransactionTime_Out_0_20_Count>0</TransactionTime_Out_0_20_Count>
      <TransactionTime_Out_20_40_Count>0</TransactionTime_Out_20_40_Count>
      <TransactionTime_Out_40_60_Count>0</TransactionTime_Out_40_60_Count>
      <TransactionTime_Out_60_80_Count>0</TransactionTime_Out_60_80_Count>
      <TransactionTime_Out_80_100_Count>0</TransactionTime_Out_80_100_Count>
      <TransactionTime_Out_100_120_Count>0</TransactionTime_Out_100_120_Count>
      <TransactionTime_Out_120_140_Count>0</TransactionTime_Out_120_140_Count>
      <TransactionTime_Out_140_160_Count>0</TransactionTime_Out_140_160_Count>
      <TransactionTime_Out_160_180_Count>0</TransactionTime_Out_160_180_Count>
      <TransactionTime_Out_180_200_Count>0</TransactionTime_Out_180_200_Count>
      <TransactionTime_Out_gt_200_Count>0</TransactionTime_Out_gt_200_Count>
    </Sample>
  </DiameterMraTdfLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterMraTdfPeerLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats> <StartTime>2001-12-31T12:00:00</StartTime>
<EndTime>2001-12-31T12:15:00</EndTime>
 <DiameterMraTdfPeerLatencyStats>
   <Name>NetworkElementName</Name>
   <NeId>NeId</NeId>
 </DiameterMraTdfPeerLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraTdfPeerLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<Statistics>
  <DiameterMraTdfPeerLatencyStats>
    <Sample>
      <StartTime>2012-05-28T09:30:00Z</StartTime>
      <EndTime>2012-05-28T09:45:00Z</EndTime>
      <MRA>kli-66-mra</MRA>
      <IsComplete>true</IsComplete>
      <Name>kli-63-mpe</Name>
      <NeId/>
      <NetworkElementType/>
      <NetworkElementSubType/>
      <ConnectAddress>10.60.25.64:3868,10.60.25.65:3868</ConnectAddress>
      <ConnectPort>0</ConnectPort>
      <ConnectType>SCTP</ConnectType>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_In_0_20_Count>0</TransactionTime_In_0_20_Count>
      <TransactionTime_In_20_40_Count>0</TransactionTime_In_20_40_Count>
      <TransactionTime_In_40_60_Count>0</TransactionTime_In_40_60_Count>
      <TransactionTime_In_60_80_Count>0</TransactionTime_In_60_80_Count>
```

```
      <TransactionTime_In_80_100_Count>0</TransactionTime_In_80_100_Count>
      <TransactionTime_In_100_120_Count>0</TransactionTime_In_100_120_Count>
      <TransactionTime_In_120_140_Count>0</TransactionTime_In_120_140_Count>
      <TransactionTime_In_140_160_Count>0</TransactionTime_In_140_160_Count>
      <TransactionTime_In_160_180_Count>0</TransactionTime_In_160_180_Count>
      <TransactionTime_In_180_200_Count>0</TransactionTime_In_180_200_Count>
      <TransactionTime_In_gt_200_Count>0</TransactionTime_In_gt_200_Count>
      <TransactionTime_Out_0_20_Count>0</TransactionTime_Out_0_20_Count>
      <TransactionTime_Out_20_40_Count>0</TransactionTime_Out_20_40_Count>
      <TransactionTime_Out_40_60_Count>0</TransactionTime_Out_40_60_Count>
      <TransactionTime_Out_60_80_Count>0</TransactionTime_Out_60_80_Count>
      <TransactionTime_Out_80_100_Count>0</TransactionTime_Out_80_100_Count>
      <TransactionTime_Out_100_120_Count>0</TransactionTime_Out_100_120_Count>
      <TransactionTime_Out_120_140_Count>0</TransactionTime_Out_120_140_Count>
      <TransactionTime_Out_140_160_Count>0</TransactionTime_Out_140_160_Count>
      <TransactionTime_Out_160_180_Count>0</TransactionTime_Out_160_180_Count>
      <TransactionTime_Out_180_200_Count>0</TransactionTime_Out_180_200_Count>
      <TransactionTime_Out_gt_200_Count>0</TransactionTime_Out_gt_200_Count>
    </Sample>
  </DiameterMraTdfPeerLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterMraDrmaLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterMraDrmaLatencyStats>
        <MRA>MRA</MRA>
      </DiameterMraDrmaLatencyStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraDrmaLatencyStats** tag defined in the XSDs. he following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterMraDrmaLatencyStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <MRA>MRA</MRA>
      <IsComplete>true</IsComplete>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
```

```
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterMraDrmaLatencyStats>
</Statistics>
```

Request

This request follows the **QueryOmStats ➤ DiameterDrmaPeerLatencyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <DiameterDrmaPeerLatencyStats >
        <Name>NetworkElement</Name>
       <NeId>NeId</NeId>
      </DiameterDrmaPeerLatencyStats >
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterDrmaPeerLatencyStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterDrmaPeerLatencyStats >
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <MRA>MRA</MRA>
      <IsComplete>true</IsComplete>
      <Name>NetworkElement</Name>
      <NeId>NeId</NeId>
      <NetworkElementType>NetworkElementType</NetworkElementType>
      <NetworkElementSubType>NetworkElementSubType</NetworkElementSubType>
      <ConnectAddress>10.60.4.56</ConnectAddress>
      <ConnectPort>3868</ConnectPort>
      <MaxTransactionInProcessingTime>0</MaxTransactionInProcessingTime>
      <AverageTransactionInProcessingTime>0</AverageTransactionInProcessingTime>
      <MaxTransactionOutProcessingTime>0</MaxTransactionOutProcessingTime>
      <AverageTransactionOutProcessingTime>0</AverageTransactionOutProcessingTime>
      <TransactionTime_0_20_Count>0</TransactionTime_0_20_Count>
      <TransactionTime_20_40_Count>0</TransactionTime_20_40_Count>
      <TransactionTime_40_60_Count>0</TransactionTime_40_60_Count>
      <TransactionTime_60_80_Count>0</TransactionTime_60_80_Count>
      <TransactionTime_80_100_Count>0</TransactionTime_80_100_Count>
      <TransactionTime_100_120_Count>0</TransactionTime_100_120_Count>
      <TransactionTime_120_140_Count>0</TransactionTime_120_140_Count>
      <TransactionTime_140_160_Count>0</TransactionTime_140_160_Count>
      <TransactionTime_160_180_Count>0</TransactionTime_160_180_Count>
      <TransactionTime_180_200_Count>0</TransactionTime_180_200_Count>
      <TransactionTime_gt_200_Count>0</TransactionTime_gt_200_Count>
    </Sample>
  </DiameterDrmaPeerLatencyStats >
</Statistics>
```

Individual statistics are defined as follows for the AF, PCEF, DRMA, and SH protocols:

- **Maximum Transaction Time**: Maximum transaction processing time.

- **Average Transaction Time**: Average transaction processing time.
- **Transactions Processed in [0-20] ms**: Number of transactions processed in 0-20 ms.
- **Transactions Processed in [20-40] ms**: Number of transactions processed in 20-40 ms.
- **Transactions Processed in [40-60] ms**: Number of transactions processed in 40-60 ms.
- **Transactions Processed in [60-80] ms**: Number of transactions processed in 60-80 ms.
- **Transactions Processed in [80-100] ms**: Number of transactions processed in 80-100 ms.
- **Transactions Processed in [100-120] ms**: Number of transactions processed in 100-120 ms.
- **Transactions Processed in [120-140] ms**: Number of transactions processed in 120-140 ms.
- **Transactions Processed in [140-160] ms**: Number of transactions processed in 140-160 ms.
- **Transactions Processed in [160-180] ms**: Number of transactions processed in 160-180 ms.
- **Transactions Processed in [180-200] ms**: Number of transactions processed in 180-200 ms.
- **Transactions Processed in [>200] ms**: Number of transactions processed in >200 ms.

RADIUS Operational Measurement Requests

This section shows the requests and responses for:

- RADIUS Accounting Statistics
- RADIUS Accounting Network Element Statistics
- RADIUS-S Statistics
- RADIUS-S Network Element Statistics

RADIUS Accounting Statistics

The following examples show the request and response that are defined in the XSDs for the RadiusAccountingStats tag.

Request

This request follows the **QueryOmStats ➤ RadiusAccountingStats** tag defined in the XSDs.

The following is an example of a request for all policy servers in the system:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <QueryOmStats>
    <StartTime>2001-12-31T12:00:00</StartTime>
    <EndTime>2001-12-31T13:00:00</EndTime>
   - <RadiusAccountingStats>
      <PolicyServer>mpe85</PolicyServer>
    </RadiusAccountingStats>
  </QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ RadiusAccountingStats** tag defined in the XSDs.

- MessagesInCount — Number of messages in.
- MessagesOutCount — Number of messages out.
- AccountingStartCount — Number of Accounting Start messages received.
- AccountingStopCount — Number of Accounting Stop messages received.
- AccountingUpdateCount — Number of Accounting Interim Update messages received.

- DuplicateMessageCount — Number of duplicate messages received.
- MD5MismatchReceivedCount — Number of messages disgarded because of MD5 mismatch errors.
- ActiveSessionCount — Number of active sessions.
- UnknownMessageCount — Number of unknown messages received.
- MaximumActiveSessionsCount — Number of maximum sessions.
- StaleSessionsCount — Number of stale sessions.

The following is an example XML response to a RadiusAccountingStats tag request:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <Statistics>
- <RadiusAccountingStats>
 - <Sample>
  <StartTime>2001-12-31T12:00:00Z</StartTime>
  <EndTime>2001-12-31T12:05:00Z</EndTime>
  <PolicyServer>mpe85</PolicyServer>
  <IsComplete>true</IsComplete>
  <MessagesInCount>0</MessagesInCount>
  <MessagesOutCount>0</MessagesOutCount>
  <AccountingStartCount>0</AccountingStartCount>
  <AccountingStopCount>0</AccountingStopCount>
  <AccountingUpdateCount>0</AccountingUpdateCount>
  <DuplicateMessageCount>0</DuplicateMessageCount>
  <Md5MismatchReceivedCount>0</Md5MismatchReceivedCount>
  <UnknownMessageCount>0</UnknownMessageCount>
  <ActiveSessionsCount>0</ActiveSessionsCount>
  <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
  <StaleSessionsCount>0</StaleSessionsCount>
 </Sample>
</RadiusAccountingStats>
</Statistics>
```

RADIUS Accounting Network Element Statistics

The following examples show the request and response that are defined in the XSDs for the RadiusAccountingNetworkElementStats tag.

Request

This request follows the **QueryOmStats ➤ RadiusAccountingNetworkElementStats** tag defined in the XSDs.

The following is an example of a request for all policy servers in the system:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <QueryOmStats>
  <StartTime>2001-12-31T12:00:00</StartTime>
  <EndTime>2001-12-31T13:00:00</EndTime>
  - <RadiusAccountingNetworkElementStats>
   <Name>Server1</Name>
  </RadiusAccountingNetworkElementStats>
  </QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ RadiusAccountingNetworkElementStats** tag defined in the XSDs.

- Name — Unique name identifying the Network Element for the following statistics.

- NeId — Optional identifier field for the Network Element.
- NetworkElementType — Type of the Network Element.
- NetworkElementSubType — Sub Type of the Network Element.
- AccountingStartCount — Number of Accounting Start messages received.
- AccountingStopCount — Number of Accounting Stop messages received.
- AccountingUpdateCount — Number of Accounting Interim Update messages received.
- DuplicateMessageCount — Number of duplicate messages received.
- MD5MismatchReceivedCount — Number of messages disgarded because of MD5 mismatch errors.
- ActiveSessionCount — Number of active sessions.
- UnknownMessageCount — Number of unknown messages received.
- MaximumActiveSessionsCount — Number of maximum sessions.
- StaleSessionsCount — Number of stale sessions.

The following is an example XML response to a RadiusAccountingNetworkElementStats tag request:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <Statistics>
- <RadiusAccountingNetworkElementStats>
 - <Sample>
  <StartTime>2001-12-31T12:00:00Z</StartTime>
  <EndTime>2001-12-31T12:05:00Z</EndTime>
  <PolicyServer>mpe85</PolicyServer>
  <IsComplete>true</IsComplete>
  <Name>Server1</Name>
  <NeId />
  <NetworkElementType>NAS</NetworkElementType>
  <NetworkElementSubType />
  <MessagesInCount>0</MessagesInCount>
  <MessagesOutCount>0</MessagesOutCount>
  <MessagesDecodedCount>0</MessagesDecodedCount>
  <AccountingStartCount>0</AccountingStartCount>
  <AccountingStopCount>0</AccountingStopCount>
  <AccountingUpdateCount>0</AccountingUpdateCount>
  <DuplicateMessageCount>0</DuplicateMessageCount>
  <Md5MismatchReceivedCount>0</Md5MismatchReceived
Count>
  <UnknownMessageCount>0</UnknownMessageCount>
  <ActiveSessionsCount>0</ActiveSessionsCount>
  <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
  <StaleSessionsCount>0</StaleSessionsCount>
 </Sample>
</RadiusAccountingNetworkElementsStats>
</Statistics>
```

RADIUS-S Statistics

The following examples show the request and response that are defined in the XSDs for the RadiusStats tag.

Request

This request follows the **QueryOmStats ➤ RadiusStats** tag defined in the XSDs.

The following is an example of a request for all policy servers in the system:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2007-10-26T14:30:00Z</StartTime>
    <EndTime>2007-10-26T14:40:00Z</EndTime>
    <RadiusStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ RadiusStats** tag defined in the XSDs.

- MessagesInCount — Number of messages in.
- MessagesOutCount — Number of messages out.
- AccountingStartCount — Number of Accounting Start messages received.
- AccountingStopCount — Number of Accounting Stop messages received.
- AccountingUpdateCount — Number of Accounting Interim Update messages received.
- AccountingResponseCount — Number of Accounting Responses sent.
- AccessRequestCount — Number of Access-Request messages received.
- AccessAcceptCount — Number of Access-Accept messages sent.
- AccessRejectCount — Number of Access-Reject messages sent.
- ReceivedMd5ErrorCount — Number of received messages dropped because of MD5 errors.
- SendMd5ErrorCount — Number of CoA requests that failed because of MD5 errors.
- ReceivedErrorCount — Number of received messages dropped because of errors.
- CoACount — Number of CoA messages sent.
- CoASuccessCount — Number of CoA-ACK messages received.
- CoAProvisionCount — Number of CoA messages to provision a default QOS profile.
- CoaApplicationCount — Number of CoA messages to adjust QOS for an application request.
- CoANckCount — Number of CoA-NAK messages received.
- NoResponseCount — Number of requests that received no response.
- UnknownGatewayRequestCount — Number of received messages dropped from unknown gateways.
- ResendCount — Number of CoA requests that were retransmitted.
- SendErrorCount — Number of CoA requests that were not sent because of errors.

The following is an example XML response to a RadiusStats tag request:

```
<?xml version="1.0" ?>
<Statistics>
    <RadiusStats>
     <Sample>
       <StartTime>2007-01-29T17:32:59Z</StartTime>
       <EndTime>2007-01-29T17:35:37Z</EndTime>
       <PolicyServer>localhost</PolicyServer>
       <IsComplete>true</IsComplete>
       <MessagesInCount>0</MessagesInCount>
       <MessagesOutCount>0</MessagesOutCount>
       <AccountingStartCount>0</AccountingStartCount>
       <AccountingStopCount>0</AccountingStopCount>
       <AccountingUpdateCount>0</AccountingUpdateCount>
       <AccountingResponseCount>0</AccountingResponseCount>
       <AccessRequestCount>0</AccessRequestCount>
       <AccessAcceptCount>0</AccessAcceptCount>
       <AccessRejectCount>0</AccessRejectCount>
```

```
      <ReceivedMd5ErrorCount>0</ReceivedMd5ErrorCount>
      <SendMd5ErrorCount>0</SendMd5ErrorCount>
      <ReceivedErrorCount>0</ReceivedErrorCount>
      <CoACount>0</CoACount>
      <CoASuccessCount>0</CoASuccessCount>
      <CoAProvisionCount>0</CoAProvisionCount>
      <CoaApplicationCount>0</CoaApplicationCount>
      <CoANckCount>0</CoANckCount>
      <NoResponseCount>0</NoResponseCount>
      <UnknownGatewayRequestCount>0</UnknownGatewayRequestCount>
      <ResendCount>0</ResendCount>
      <SendErrorCount>0</SendErrorCount>
    </Sample>
    <Sample>
      <StartTime>2007-01-29T17:35:37Z</StartTime>
      <EndTime>2007-01-29T17:37:13Z</EndTime>
      <PolicyServer>localhost</PolicyServer>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <AccountingStartCount>0</AccountingStartCount>
      <AccountingStopCount>0</AccountingStopCount>
      <AccountingUpdateCount>0</AccountingUpdateCount>
      <AccountingResponseCount>0</AccountingResponseCount>
      <AccessRequestCount>0</AccessRequestCount>
      <AccessAcceptCount>0</AccessAcceptCount>
      <AccessRejectCount>0</AccessRejectCount>
      <ReceivedMd5ErrorCount>0</ReceivedMd5ErrorCount>
      <SendMd5ErrorCount>0</SendMd5ErrorCount>
      <ReceivedErrorCount>0</ReceivedErrorCount>
      <CoACount>0</CoACount>
      <CoASuccessCount>0</CoASuccessCount>
      <CoAProvisionCount>0</CoAProvisionCount>
      <CoaApplicationCount>0</CoaApplicationCount>
      <CoANckCount>0</CoANckCount>
      <NoResponseCount>0</NoResponseCount>
      <UnknownGatewayRequestCount>0</UnknownGatewayRequestCount>
      <ResendCount>0</ResendCount>
      <SendErrorCount>0</SendErrorCount>
     </Sample>
     <Sample>
      <StartTime>2007-01-29T17:37:13Z</StartTime>
      <EndTime>2007-01-29T17:45:00Z</EndTime>
      <PolicyServer>localhost</PolicyServer>
      <IsComplete>true</IsComplete>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <AccountingStartCount>0</AccountingStartCount>
      <AccountingStopCount>0</AccountingStopCount>
      <AccountingUpdateCount>0</AccountingUpdateCount>
      <AccountingResponseCount>0</AccountingResponseCount>
      <AccessRequestCount>0</AccessRequestCount>
      <AccessAcceptCount>0</AccessAcceptCount>
      <AccessRejectCount>0</AccessRejectCount>
      <ReceivedMd5ErrorCount>0</ReceivedMd5ErrorCount>
      <SendMd5ErrorCount>0</SendMd5ErrorCount>
      <ReceivedErrorCount>0</ReceivedErrorCount>
      <CoACount>0</CoACount>
      <CoASuccessCount>0</CoASuccessCount>
      <CoAProvisionCount>0</CoAProvisionCount>
      <CoaApplicationCount>0</CoaApplicationCount>
      <CoANckCount>0</CoANckCount>
      <NoResponseCount>0</NoResponseCount>
      <UnknownGatewayRequestCount>0</UnknownGatewayRequestCount>
      <ResendCount>0</ResendCount>
      <SendErrorCount>0</SendErrorCount>
```

```
     </Sample>
     <Sample>
       <StartTime>2007-01-29T18:30:00Z</StartTime>
       <EndTime>2007-01-29T18:45:00Z</EndTime>
       <PolicyServer>localhost</PolicyServer>
       <IsComplete>true</IsComplete>
       <MessagesInCount>0</MessagesInCount>
       <MessagesOutCount>0</MessagesOutCount>
       <AccountingStartCount>0</AccountingStartCount>
       <AccountingStopCount>0</AccountingStopCount>
       <AccountingUpdateCount>0</AccountingUpdateCount>
       <AccountingResponseCount>0</AccountingResponseCount>
       <AccessRequestCount>0</AccessRequestCount>
       <AccessAcceptCount>0</AccessAcceptCount>
       <AccessRejectCount>0</AccessRejectCount>
       <ReceivedMd5ErrorCount>0</ReceivedMd5ErrorCount>
       <SendMd5ErrorCount>0</SendMd5ErrorCount>
       <ReceivedErrorCount>0</ReceivedErrorCount>
       <CoACount>0</CoACount>
       <CoASuccessCount>0</CoASuccessCount>
       <CoAProvisionCount>0</CoAProvisionCount>
       <CoaApplicationCount>0</CoaApplicationCount>
       <CoANckCount>0</CoANckCount>
       <NoResponseCount>0</NoResponseCount>
       <UnknownGatewayRequestCount>0</UnknownGatewayRequestCount>
       <ResendCount>0</ResendCount>
       <SendErrorCount>0</SendErrorCount>
     </Sample>
    </RadiusStats>
</Statistics>
```

RADIUS-S Network Element Statistics

The following examples show the request and response that are defined in the XSDs for the RadiusNetworkElementStats tag.

Request

This request follows the **QueryOmStats ➤ RadiusNetworkElementStats** tag defined in the XSDs.

The following is an example of a request for a single network element using the Name parameter:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:35:00Z</EndTime>
    <RadiusNetworkElementStats>
        <Name>Router1</Name>
    </RadiusNetworkElementStats>
</QueryOmStats>
```

The following is an example of a request for multiple network elements using the Name(s) and NeId(s) parameters. This example returns statisitics for three different network elements:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2006-10-12T11:15:00Z</StartTime>
    <EndTime>2006-10-12T11:25:00Z</EndTime>
    <RadiusNetworkElementStats>
        <Name>Router1</Name>
        <Name>Router2</Name>
        <Name>Router3</Name>
```

```
    </RadiusNetworkElementStats>
</QueryOmStats>
```

The following is an example of a request for all network elements in the system. This example returns statistics for each network element:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2006-10-12T11:15:00Z</StartTime>
    <EndTime>2006-10-12T11:25:00Z</EndTime>
    <RadiusNetworkElementStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ RadiusNetworkElementStats** tag defined in the XSDs.

Both Name and NeId are returned along with the statistics and actual recorded times for those statistics. Sample groups are ordered by policy server, network element, and then by time.

- Name — Unique name identifying the Network Element for the following statistics.
  - NeId — Optional identifier field for the Network Element.
  - NetworkElementType — Type of the Network Element.
  - NetworkElementSubType — Sub Type of the Network Element.
  - AccountingStartCount — Number of Accounting Start messages received.
  - AccountingStopCount — Number of Accounting Stop messages received.
  - AccountingUpdateCount — Number of Accounting Interim Update messages received.
  - AccountingResponseCount — Number of Accounting responses sent.
  - AccessRequestCount — Number of Access-Request messages received.
  - AccessAcceptCount — Number of Access-Accept messages received.
  - AccessRejectCount — Number of Access-Reject messages received.
  - ReceivedMd5ErrorCount — Number of received messages dropped because of MD5 errors.
  - SendMd5ErrorCount — Number of CoA requests that failed because of MD5 errors.
  - ReceivedErrorCount — Number of messages dropped because of errors.
  - CoACount — Number of CoA messages sent.
  - CoASuccessCount — Number of CoA-ACK messages received.
  - CoAProvisionCount — Number of CoA messages to provision a default QoS profile.
  - CoaApplicationCount — Number of CoA messages to adjust QoS for an application request.
  - CoANckCount — Number of CoA-NAK messages received.
  - NoResponseCount — Number of CoA requests that received no response.
  - SendErrorCount — Number of CoA requests that were not sent because of errors.
  - ResendCount — Number of CoA requests that were retransmitted.

The following is an example XML response to a multiple network element request:

```
<?xml version="1.0" ?>
<Statistics>
    <RadiusNetworkElementStats>
        <Sample>
            <StartTime>2007-01-29T17:32:59Z</StartTime>
            <EndTime>2007-01-29T17:35:37Z</EndTime>
            <PolicyServer>localhost</PolicyServer>
```

```
            <Name>simPDSN</Name>
            <NeId/>
            <NetworkElementType>PDSN</NetworkElementType>
            <NetworkElementSubType>Starent</NetworkElementSubType>
            <AccountingStartCount>0</AccountingStartCount>
            <AccountingStopCount>0</AccountingStopCount>
            <AccountingUpdateCount>0</AccountingUpdateCount>
            <AccountingResponseCount>0</AccountingResponseCount>
            <AccessRequestCount>0</AccessRequestCount>
            <AccessAcceptCount>0</AccessAcceptCount>
            <AccessRejectCount>0</AccessRejectCount>
            <ReceivedMd5ErrorCount>0</ReceivedMd5ErrorCount>
            <SendMd5ErrorCount>0</SendMd5ErrorCount>
            <ReceivedErrorCount>0</ReceivedErrorCount>
            <CoACount>0</CoACount>
            <CoASuccessCount>0</CoASuccessCount>
            <CoAProvisionCount>0</CoAProvisionCount>
            <CoaApplicationCount>0</CoaApplicationCount>
            <CoANckCount>0</CoANckCount>
            <NoResponseCount>0</NoResponseCount>
            <SendErrorCount>0</SendErrorCount>
            <ResendCount>0</ResendCount>
        </Sample>
        <Sample>
            <StartTime>2007-01-29T17:35:37Z</StartTime>
            <EndTime>2007-01-29T17:37:13Z</EndTime>
            <PolicyServer>localhost</PolicyServer>
            <Name>simPDSN</Name>
            <NeId/>
            <NetworkElementType>PDSN</NetworkElementType>
            <NetworkElementSubType>Starent</NetworkElementSubType>
            <AccountingStartCount>0</AccountingStartCount>
            <AccountingStopCount>0</AccountingStopCount>
            <AccountingUpdateCount>0</AccountingUpdateCount>
            <AccountingResponseCount>0</AccountingResponseCount>
            <AccessRequestCount>0</AccessRequestCount>
            <AccessAcceptCount>0</AccessAcceptCount>
            <AccessRejectCount>0</AccessRejectCount>
            <ReceivedMd5ErrorCount>0</ReceivedMd5ErrorCount>
            <SendMd5ErrorCount>0</SendMd5ErrorCount>
            <ReceivedErrorCount>0</ReceivedErrorCount>
            <CoACount>0</CoACount>
            <CoASuccessCount>0</CoASuccessCount>
            <CoAProvisionCount>0</CoAProvisionCount>
            <CoaApplicationCount>0</CoaApplicationCount>
            <CoANckCount>0</CoANckCount>
            <NoResponseCount>0</NoResponseCount>
            <SendErrorCount>0</SendErrorCount>
            <ResendCount>0</ResendCount>
        </Sample>
        <Sample>
            <StartTime>2007-01-29T17:37:13Z</StartTime>
            <EndTime>2007-01-29T17:45:00Z</EndTime>
            <PolicyServer>localhost</PolicyServer>
            <Name>simPDSN</Name>
            <NeId/>
            <NetworkElementType>PDSN</NetworkElementType>
            <NetworkElementSubType>Starent</NetworkElementSubType>
            <AccountingStartCount>0</AccountingStartCount>
            <AccountingStopCount>0</AccountingStopCount>
            <AccountingUpdateCount>0</AccountingUpdateCount>
            <AccountingResponseCount>0</AccountingResponseCount>
            <AccessRequestCount>0</AccessRequestCount>
            <AccessAcceptCount>0</AccessAcceptCount>
            <AccessRejectCount>0</AccessRejectCount>
```

```
            <ReceivedMd5ErrorCount>0</ReceivedMd5ErrorCount>
            <SendMd5ErrorCount>0</SendMd5ErrorCount>
            <ReceivedErrorCount>0</ReceivedErrorCount>
            <CoACount>0</CoACount>
            <CoASuccessCount>0</CoASuccessCount>
            <CoAProvisionCount>0</CoAProvisionCount>
            <CoaApplicationCount>0</CoaApplicationCount>
            <CoANckCount>0</CoANckCount>
            <NoResponseCount>0</NoResponseCount>
            <SendErrorCount>0</SendErrorCount>
            <ResendCount>0</ResendCount>
        </Sample>
        <Sample>
            <StartTime>2007-01-29T17:45:00Z</StartTime>
            <EndTime>2007-01-29T18:00:01Z</EndTime>
            <PolicyServer>localhost</PolicyServer>
            <Name>simPDSN</Name>
            <NeId/>
            <NetworkElementType>PDSN</NetworkElementType>
            <NetworkElementSubType>Starent</NetworkElementSubType>
            <AccountingStartCount>0</AccountingStartCount>
            <AccountingStopCount>0</AccountingStopCount>
            <AccountingUpdateCount>0</AccountingUpdateCount>
            <AccountingResponseCount>0</AccountingResponseCount>
            <AccessRequestCount>0</AccessRequestCount>
            <AccessAcceptCount>0</AccessAcceptCount>
            <AccessRejectCount>0</AccessRejectCount>
            <ReceivedMd5ErrorCount>0</ReceivedMd5ErrorCount>
            <SendMd5ErrorCount>0</SendMd5ErrorCount>
            <ReceivedErrorCount>0</ReceivedErrorCount>
            <CoACount>0</CoACount>
            <CoASuccessCount>0</CoASuccessCount>
            <CoAProvisionCount>0</CoAProvisionCount>
            <CoaApplicationCount>0</CoaApplicationCount>
            <CoANckCount>0</CoANckCount>
            <NoResponseCount>0</NoResponseCount>
            <SendErrorCount>0</SendErrorCount>
            <ResendCount>0</ResendCount>
        </Sample>
    </RadiusNetworkElementStats>
</Statistics>
```

Connected Network Elements Statistics

The following examples show the request and response that are defined in the XSDs for the ConnectedNetworkElementStats tag.

Request

This request follows the **QueryOmStats ➤ ConnectedNetworkElementStats** tag defined in the XSDs.

The following is an example of a request for all policy servers in the system:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2007-10-26T14:30:00Z</StartTime>
    <EndTime>2007-10-26T14:40:00Z</EndTime>
```

```
    <ConnectedNetworkElementStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ ConnectedNetworkElementStats** tag defined in the XSDs.

ConnectedNetworkElementCount — The absolute number of network elements maintaining a consistent connection to each MPE. For example, B-RAS elements.

The following is an example XML response to a ConnectedNetworkElementStats tag request:

```
<xml version="1.0" encoding="UTF-8"?>
<Statistics>
    <ConnectedNetworkElementStats>
        <Sample>
            <StartTime>2007-10-26T14:30:22Z</StartTime>
            <EndTime>2007-10-26T14:35:11Z</EndTime>
            <PolicyServer>Atlanta105</PolicyServer>
            <ConnectedNetworkElementCount>52</ConnectedNetworkElementCount>
        </Sample>
    </ConnectedNetworkElementStats>
</Statistics>
```

Mgpi Statistics

The following examples show the request and response that are defined in the XSDs for the MgpiStats tag.

Request

This request follows the **QueryOmStats ➤ MgpiStats** tag defined in the XSDs.

The following is an example of a request for all policy servers within the system:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2007-10-26T14:30:00Z</StartTime>
    <EndTime>2007-10-26T14:40:00Z</EndTime>
    <MgpiStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ MgpiStats** tag defined in the XSDs.

- TotalFlowsCount — The sum of GPI across all PCMM sessions created from the MGPI path.
- ActualGatesCount — The actual number of PCMM gates/sessions created from the MGPI path.
- MultiFlowGatesCount — The number of PCMM gates/sessions where the GPI is greater then 1.
- EffectiveGatesCount — The number of PCMM gates/sessions, less the number of Actual Gates and TotalFlows.

The following is an example XML response to a MgpiStats tag request:

```
<xml version="1.0" encoding="UTF-8"?>
<Statistics>
```

```
    <MgpiStats>
        <Sample>
         <StartTime>2007-10-26T14:30:22Z</StartTime>
         <EndTime>2007-10-26T14:35:11Z</EndTime>
            <PolicyServer>Atlanta105</PolicyServer>
            <IsComplete>true</IsComplete>
            <TotalFlowsCount>10</TotalFlowsCount>
            <AcutalGatesCount>52</ActualGatesCount>
         <MultiFlowGatesCount>52</MultiFlowGatesCount>
         <EffectiveGatesCount>15</EffectiveGatesCount>
        </Sample>
    </MgpiStats>
</Statistics>
```

Policy Statistics

The following examples show the request and response that are defined in the XSDs for the PolicyStats tag.

Request

This request follows the **QueryOmStats ➤ PolicyStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
  <StartTime>2001-12-31T12:00:00</StartTime>
  <EndTime>2001-12-31T12:15:00</EndTime>
  <PolicyStats>
    <PolicyServer>PolicyServer</PolicyServer>
    <Name>PolicyName</Name>
  </PolicyStats>
</QueryOmStats>
```

The following is an example of a request for all policy servers in the system:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
   <StartTime>2006-10-26T14:30:00Z</StartTime>
   <EndTime>2006-10-26T14:40:00Z</EndTime>
 <PolicyStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ PolicyStats** tag defined in the XSDs.

- Total execution time — the summary of all execution durations, where execution duration is measured starting the beginning of the policy conditions evaluation until the execution finishing.
- Maximum execution time — the longest execution duration of the policy.
- Average execution time — the arithmetic average off all execution durations of the policy.
- TriggerCount — number of policies triggered.
- FailCondCount — number of policies that failed because of conditions.
- FailNACount — number of policies that failed because of data not available/ applicable.
- FailExcCount — number of policies that failed because of exceptions.

The following is an example XML response to a PolicyStats tag request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <PolicyStats>
    <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <IsComplete>true</IsComplete>
         <Name>PolicyName</Name>
         <EvalCount>0</EvalCount>
         <TriggerCount>0</TriggerCount>
         <FailCondCount>0</FailCondCount>
         <FailNACount>0</FailNACount>
         <FailExcCount>0</FailExcCount>
         <TotalExecTime>0</TotalExecTime>
         <AvgExecTime>0</AvgExecTime>
         <MaxExecTime>0</MaxExecTime>
         <MaxExecTime>0</MaxExecTime>
         <HistExecTime_000_020>0</HistExecTime_000_020>
         <HistExecTime_020_040>0</HistExecTime_020_040>
         <HistExecTime_040_060>0</HistExecTime_040_060>
         <HistExecTime_060_080>0</HistExecTime_060_080>
         <HistExecTime_080_100>0</HistExecTime_080_100>
         <HistExecTime_100_150>0</HistExecTime_100_150>
         <HistExecTime_150_200>0</HistExecTime_150_200>
         <HistExecTime_200_250>0</HistExecTime_200_250>
         <HistExecTimePlus_250>0</HistExecTimePlus_250>
     </PolicyStats>
<Statistics>
```

Diameter Operational Measurements Requests

This section shows the requests and responses for:

- *Diameter Application Function Statistics*
- *Diameter Application Function Peer Statistics*
- *Diameter Charging Function Statistics*
- *Diameter Charging Function Peer Statistics*
- *Diameter Policy Charging Enforcement Function Statistics*
- *Diameter Policy Charging Enforcement Function Peer Statistics*

Diameter Sh Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterShStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterShStats** tag defined in the XSDs.

The following is an example of a request for all policy servers within the system:

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<QueryOmStats>
   <StartTime>2001-12-31T12:00:00</StartTime>
   <EndTime>2001-12-31T13:00:00</EndTime>
   <DiameterShStats>
     <PolicyServer>mpe85</PolicyServer>
   </DiameterShStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterShStats** tag defined in the XSDs.

The following is an example XML response to a DiameterShStats tag request:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Statistics>
   <DiameterShStats>
   <Sample>
   <StartTime>2001-12-31T12:00:00Z</StartTime>
   <EndTime>2001-12-31T12:15:00Z</EndTime>
   <PolicyServer>mpe85</PolicyServer>
   <IsComplete>true</IsComplete>
   <CurrentConnectionsCount>0</CurrentConnectionsCount>
   <MessagesInCount>0</MessagesInCount>
   <MessagesOutCount>0</MessagesOutCount>
   <UDRMessagesReceivedCount>0</UDRMessagesReceivedCount>
   <UDRMessagesSentCount>0</UDRMessagesSentCount>
   <UDASuccessMessagesReceivedCount>0</UDASuccessMessagesReceivedCount>
   <UDASuccessMessagesSentCount>0</UDASuccessMessagesSentCount>
   <UDAFailureMessagesReceivedCount>0</UDAFailureMessagesReceivedCount>
   <UDAFailureMessagesSentCount>0</UDAFailureMessagesSentCount>
   <PNRMessagesReceivedCount>0</PNRMessagesReceivedCount>
   <PNRMessagesSentCount>0</PNRMessagesSentCount>
   <PNASuccessMessagesReceivedCount>0</PNASuccessMessagesReceivedCount>
   <PNASuccessMessagesSentCount>0</PNASuccessMessagesSentCount>
   <PNAFailureMessagesReceivedCount>0</PNAFailureMessagesReceivedCount>
   <PNAFailureMessagesSentCount>0</PNAFailureMessagesSentCount>
   <PURMessagesReceivedCount>0</PURMessagesReceivedCount>
   <PURMessagesSentCount>0</PURMessagesSentCount>
   <PUASuccessMessagesReceivedCount>0</PUASuccessMessagesReceivedCount>
   <PUASuccessMessagesSentCount>0</PUASuccessMessagesSentCount>
   <PUAFailureMessagesReceivedCount>0</PUAFailureMessagesReceivedCount>
   <PUAFailureMessagesSentCount>0</PUAFailureMessagesSentCount>
   <SNRMessagesReceivedCount>0</SNRMessagesReceivedCount>
   <SNRMessagesSentCount>0</SNRMessagesSentCount>
   <SNASuccessMessagesReceivedCount>0</SNASuccessMessagesReceivedCount>
   <SNASuccessMessagesSentCount>0</SNASuccessMessagesSentCount>
   <SNAFailureMessagesReceivedCount>0</SNAFailureMessagesReceivedCount>
   <SNAFailureMessagesSentCount>0</SNAFailureMessagesSentCount>
   <ActiveSessionsCount>0</ActiveSessionsCount>
   <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
   <PeerOkayCount>0</PeerOkayCount>
   <PeerDownCount>0</PeerDownCount>
   <PeerSuspectCount>0</PeerSuspectCount>
   <PeerReopenCount>0</PeerReopenCount>
  </Sample>
```

```
 </DiameterShStats>
</Statistics>
```

Diameter Sh Peer Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterShPeerStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterShPeerStats** tag defined in the XSDs.

The following is an example of a request for all policy servers within the system:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
   <StartTime>2001-12-31T12:00:00</StartTime>
   <EndTime>2001-12-31T13:00:00</EndTime>
   <DiameterShPeerStats>
     <Name>server1</Name>
   </DiameterShPeerStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterShPeerStats** tag defined in the XSDs.

The following is an example XML response to a DiameterShPeerStats tag request:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Statistics>
  <DiameterShPeerStats>
   <Sample>
   <StartTime>2001-12-31T12:00:00Z</StartTime>
   <EndTime>2001-12-31T12:15:00Z</EndTime>
   <PolicyServer>mpe15</PolicyServer>
   <IsComplete>true</IsComplete>
   <Name>Server1</Name>
   <NeId />
   <NetworkElementType />
   <NetworkElementSubType />
   <ConnectTime />
   <DisconnectTime />
   <ConnectAddress />
   <ConnectPort />
   <MessagesInCount>0</MessagesInCount>
   <MessagesOutCount>0</MessagesOutCount>
   <MessagesErrorInCount>0</MessagesErrorInCount>
   <MessagesErrorOutCount>0</MessagesErrorOutCount>
   <UDRMessagesReceivedCount>0</UDRMessagesReceivedCount>
   <UDRMessagesSentCount>0</UDRMessagesSentCount>
   <UDASuccessMessagesReceivedCount>0</UDASuccessMessagesReceivedCount>
   <UDASuccessMessagesSentCount>0</UDASuccessMessagesSentCount>
   <UDAFailureMessagesReceivedCount>0</UDAFailureMessagesReceivedCount>
   <UDAFailureMessagesSentCount>0</UDAFailureMessagesSentCount>
   <PNRMessagesReceivedCount>0</PNRMessagesReceivedCount>
   <PNRMessagesSentCount>0</PNRMessagesSentCount>
   <PNASuccessMessagesReceivedCount>0</PNASuccessMessagesReceivedCount>
   <PNASuccessMessagesSentCount>0</PNASuccessMessagesSentCount>
   <PNAFailureMessagesReceivedCount>0</PNAFailureMessagesReceivedCount>
   <PNAFailureMessagesSentCount>0</PNAFailureMessagesSentCount>
   <PURMessagesReceivedCount>0</PURMessagesReceivedCount>
```

```
   <PURMessagesSentCount>0</PURMessagesSentCount>
   <PUASuccessMessagesReceivedCount>0</PUASuccessMessagesReceivedCount>
   <PUASuccessMessagesSentCount>0</PUASuccessMessagesSentCount>
   <PUAFailureMessagesReceivedCount>0</PUAFailureMessagesReceivedCount>
   <PUAFailureMessagesSentCount>0</PUAFailureMessagesSentCount>
   <SNRMessagesReceivedCount>0</SNRMessagesReceivedCount>
   <SNRMessagesSentCount>0</SNRMessagesSentCount>
   <SNASuccessMessagesReceivedCount>0</SNASuccessMessagesReceivedCount>
   <SNASuccessMessagesSentCount>0</SNASuccessMessagesSentCount>
   <SNAFailureMessagesReceivedCount>0</SNAFailureMessagesReceivedCount>
   <SNAFailureMessagesSentCount>0</SNAFailureMessagesSentCount>
   <ActiveSessionsCount>0</ActiveSessionsCount>
   <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
  </Sample>
 </DiameterShPeerStats>
</Statistics>
```

Diameter Application Function Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterAfStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterAfStats** tag defined in the XSDs.

The following is an example of a request for all policy servers within the system:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2007-10-26T14:30:00Z</StartTime>
    <EndTime>2007-10-26T14:40:00Z</EndTime>
    <DiameterAfStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterAfStats** tag defined in the XSDs.

The following is an example XML response to a DiameterAfStats tag request:

```
<Statistics>
  <DiameterAfStats>
   <Sample>
     <StartTime>2011-09-01T00:00:02Z</StartTime>
     <EndTime>2011-09-01T00:15:02Z</EndTime>
     <PolicyServer>10.15.24.106</PolicyServer>
     <IsComplete>true</IsComplete>
     <PendingConnectionsCount>0</PendingConnectionsCount>
     <CurrentConnectionsCount>1</CurrentConnectionsCount>
     <MessagesInCount>0</MessagesInCount>
     <MessagesOutCount>0</MessagesOutCount>
     <ASRMessagesReceivedCount>0</ASRMessagesReceivedCount>
     <ASRMessagesSentCount>0</ASRMessagesSentCount>
     <ASASuccessMessagesReceivedCount>0</ASASuccessMessagesReceivedCount>
     <ASASuccessMessagesSentCount>0</ASASuccessMessagesSentCount>
     <ASAFailureMessagesReceivedCount>0</ASAFailureMessagesReceivedCount>
     <ASAFailureMessagesSentCount>0</ASAFailureMessagesSentCount>
     <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
     <RARMessagesSentCount>0</RARMessagesSentCount>
     <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
     <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
```

```
     <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
     <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
     <STRMessagesReceivedCount>0</STRMessagesReceivedCount>
     <STRMessagesSentCount>0</STRMessagesSentCount>
     <STASuccessMessagesReceivedCount>0</STASuccessMessagesReceivedCount>
     <STASuccessMessagesSentCount>0</STASuccessMessagesSentCount>
     <STAFailureMessagesReceivedCount>0</STAFailureMessagesReceivedCount>
     <STAFailureMessagesSentCount>0</STAFailureMessagesSentCount>
     <AARMessagesReceivedCount>0</AARMessagesReceivedCount>
     <AARMessagesSentCount>0</AARMessagesSentCount>
     <AARInitialMessagesReceivedCount>0</AARInitialMessagesReceivedCount>
     <AARInitialMessagesSentCount>0</AARInitialMessagesSentCount>
    <AARModificationMessagesReceivedCount>0</AARModificationMessagesReceivedCount>

     <AARModificationMessagesSentCount>0</AARModificationMessagesSentCount>
     <AAASuccessMessagesReceivedCount>0</AAASuccessMessagesReceivedCount>
     <AAASuccessMessagesSentCount>0</AAASuccessMessagesSentCount>
     <AAAFailureMessagesReceivedCount>0</AAAFailureMessagesReceivedCount>
     <AAAFailureMessagesSentCount>0</AAAFailureMessagesSentCount>
     <RxPcmmMessagesTimeoutCount>0</RxPcmmMessagesTimeoutCount>
     <ActiveSessionsCount>0</ActiveSessionsCount>
     <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
     <PeerOkayCount>1</PeerOkayCount>
     <PeerDownCount>0</PeerDownCount>
     <PeerSuspectCount>0</PeerSuspectCount>
     <PeerReopenCount>0</PeerReopenCount>
    </Sample>
  </DiameterAfStats>
</Statistics>
```

Diameter Application Function Peer Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterAfPeerStats tag.

Request

This request follows the QueryOmStats ->DiameterAfPeerStats tag defined in the XSDs.

The following is an example of a request for a single peer using the Name parameter:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:35:00Z</EndTime>
    <DiameterAfPeerStats>
        <Name>Server1</Name>
    </DiameterAfPeerStats>
</QueryOmStats>
```

The following is an example of a request for all peer elements in the system. This request returns stats for each peer element:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2007-01-29T17:30:00Z</StartTime>
    <EndTime>2006-01-29T18:10:00Z</EndTime>
    <DiameterAfPeerStats/>
</QueryOmStats>
```

Response

The response to this request follows the Statistics -> DiameterAfPeertStats tag defined in the XSDs.

Both the Name and NeId are returned, along with the statistics and actual recorded times for those statistics. Samples are ordered by policy server, network element, and then by time.

The following is an example XML response to a multiple network element tag request:

```
<Statistics>
  <DiameterAfPeerStats>
        <Sample>
           <StartTime>2011-08-19T11:15:02Z</StartTime>
           <EndTime>2011-08-19T11:30:00Z</EndTime>
           <PolicyServer>10.60.24.47</PolicyServer>
           <IsComplete>true</IsComplete>
           <Name>identity2</Name>
           <NeId></NeId>
           <NetworkElementType></NetworkElementType>
           <NetworkElementSubType></NetworkElementSubType>
           <ConnectTime></ConnectTime>
           <DisconnectTime></DisconnectTime>
           <ConnectAddress></ConnectAddress>
           <ConnectPort></ConnectPort>
           <MessagesInCount>0</MessagesInCount>
           <MessagesOutCount>0</MessagesOutCount>
           <ASRMessagesReceivedCount>0</ASRMessagesReceivedCount>
           <ASRMessagesSentCount>0</ASRMessagesSentCount>
           <ASASuccessMessagesReceivedCount>0</ASASuccessMessagesReceivedCount>
           <ASASuccessMessagesSentCount>0</ASASuccessMessagesSentCount>
           <ASAFailureMessagesReceivedCount>0</ASAFailureMessagesReceivedCount>
           <ASAFailureMessagesSentCount>0</ASAFailureMessagesSentCount>
           <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
           <RARMessagesSentCount>0</RARMessagesSentCount>
           <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
           <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
           <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
           <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
           <STRMessagesReceivedCount>0</STRMessagesReceivedCount>
           <STRMessagesSentCount>0</STRMessagesSentCount>
           <STASuccessMessagesReceivedCount>0</STASuccessMessagesReceivedCount>
           <STASuccessMessagesSentCount>0</STASuccessMessagesSentCount>
           <STAFailureMessagesReceivedCount>0</STAFailureMessagesReceivedCount>
           <STAFailureMessagesSentCount>0</STAFailureMessagesSentCount>
           <AARMessagesReceivedCount>0</AARMessagesReceivedCount>
           <AARMessagesSentCount>0</AARMessagesSentCount>
           <AARInitialMessagesReceivedCount>0</AARInitialMessagesReceivedCount>
           <AARInitialMessagesSentCount>0</AARInitialMessagesSentCount>

<AARModificationMessagesReceivedCount>0</AARModificationMessagesReceivedCount>
           <AARModificationMessagesSentCount>0</AARModificationMessagesSentCount>
           <AAASuccessMessagesReceivedCount>0</AAASuccessMessagesReceivedCount>
           <AAASuccessMessagesSentCount>0</AAASuccessMessagesSentCount>
           <AAAFailureMessagesReceivedCount>0</AAAFailureMessagesReceivedCount>
           <AAAFailureMessagesSentCount>0</AAAFailureMessagesSentCount>
           <RxPcmmMessagesTimeoutCount>0</RxPcmmMessagesTimeoutCount>
        </Sample>
   <DiameterAfPeerStats>
<Statistics>
```

Diameter Charging Function Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterCTFStats tag.

Request

This request follows the QueryOmStats ->DiameterCTFStats tag defined in the XSDs.

The following is an example of a request for all policy servers in the system:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2007-01-29T17:30:00Z</StartTime>
    <EndTime>2007-01-29T18:50:00Z</EndTime>
    <DiameterCTFStats/>
</QueryOmStats>
```

Response

The response to this request follows the Statistics -> DiameterCTFStats tag defined in the XSDs.

The following is an example XML response to a multiple network element tag request:

```
<?xml version='1.0' ?>
<Statistics>
 <DiameterCTFStats>
  <Sample>
   <StartTime>2009-08-19T19:51:28Z</StartTime>
   <EndTime>2009-08-19T19:52:47Z</EndTime>
   <PolicyServer>localps</PolicyServer>
   <IsComplete>true</IsComplete>
   <CurrentConnectionsCount>0</CurrentConnectionsCount>
   <PeerOkayCount>0</PeerOkayCount>
   <PeerDownCount>0</PeerDownCount>
   <PeerSuspectCount>0</PeerSuspectCount>
   <PeerReopenCount>0</PeerReopenCount>
   <MessagesInCount>0</MessagesInCount>
   <MessagesOutCount>0</MessagesOutCount>
   <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
   <CCRMessagesSentCount>0</CCRMessagesSentCount>
   <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
   <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
   <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
   <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
   <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
   <RARMessagesSentCount>0</RARMessagesSentCount>
   <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
   <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
   <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
   <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
   <ASRMessagesReceivedCount>0</ASRMessagesReceivedCount>
   <ASRMessagesSentCount>0</ASRMessagesSentCount>
   <ASASuccessMessagesReceivedCount>0</ASASuccessMessagesReceivedCount>
   <ASASuccessMessagesSentCount>0</ASASuccessMessagesSentCount>
   <ASAFailureMessagesReceivedCount>0</ASAFailureMessagesReceivedCount>
   <ASAFailureMessagesSentCount>0</ASAFailureMessagesSentCount>
   <ActiveSessionsCount>0</ActiveSessionsCount>
   <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
  </Sample>
    </DiameterCTFStats>
</Statistics>
```

Diameter Charging Function Peer Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterCTFPeerStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterCTFPeerStats** tag defined in the XSDs.

The following is an example of a request for a single network element using the Name parameter:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:35:00Z</EndTime>
    <DiameterCTFPeerStats>
        <Name>Server1</Name>
    </DiameterCTFPeerStats>
</QueryOmStats>
```

The following is an example of a request for all network elements in the system. This request returns statistics for each network element.

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2007-01-29T17:30:00Z</StartTime>
    <EndTime>2006-01-29T18:10:00Z</EndTime>
    <DiameterCTFPeerStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterCTFPeerStats** tag defined in the XSDs.

Both Name and NeId are returned along with the statistics and actual recorded times for those statistics. Samples are ordered by policy server, network element, and then by time.

The following is an example XML response to a multiple network element tag request:

```
<?xml version='1.0' ?>
<Statistics>
 <DiameterCTFPeerStats>
  <Sample>
   <StartTime>2009-08-19T20:08:53Z</StartTime>
   <EndTime>2009-08-19T20:15:00Z</EndTime>
   <PolicyServer>localps</PolicyServer>
   <IsComplete>true</IsComplete>
   <Name>ctf1</Name>
   <NeId></NeId>
   <NetworkElementType></NetworkElementType>
   <NetworkElementSubType></NetworkElementSubType>
   <ActiveConnectionCount>0</ActiveConnectionCount>
   <MessagesInCount>0</MessagesInCount>
   <MessagesOutCount>0</MessagesOutCount>
   <MessagesErrorInCount>0</MessagesErrorInCount>
   <MessagesErrorOutCount>0</MessagesErrorOutCount>
   <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
   <CCRMessagesSentCount>0</CCRMessagesSentCount>
   <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
   <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
   <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
   <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
   <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
   <RARMessagesSentCount>0</RARMessagesSentCount>
   <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
   <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
   <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
   <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
```

```
   <ASRMessagesReceivedCount>0</ASRMessagesReceivedCount>
   <ASRMessagesSentCount>0</ASRMessagesSentCount>
   <ASASuccessMessagesReceivedCount>0</ASASuccessMessagesReceivedCount>
   <ASASuccessMessagesSentCount>0</ASASuccessMessagesSentCount>
   <ASAFailureMessagesReceivedCount>0</ASAFailureMessagesReceivedCount>
   <ASAFailureMessagesSentCount>0</ASAFailureMessagesSentCount>
   <ActiveSessionsCount>0</ActiveSessionsCount>
   <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
  </Sample>
 </DiameterCTFPeerStats>
</Statistics>
```

Diameter Policy Charging Enforcement Function Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterPcefStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterPcefStats** tag defined in the XSDs.

The following is an example of a request for all policy servers in the system.

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2007-01-29T17:30:00Z</StartTime>
    <EndTime>2006-01-29T18:10:00Z</EndTime>
    <DiameterPcefStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterPcefStats** tag defined in the XSDs.

The following is an example XML response to a DiameterPcefStats tag request:

```
<?xml version='1.0' ?>
<Statistics>
 <DiameterPcefStats>
  <Sample>
   <StartTime>2009-08-19T19:51:28Z</StartTime>
   <EndTime>2009-08-19T19:52:47Z</EndTime>
   <PolicyServer>localps</PolicyServer>
   <IsComplete>true</IsComplete>
   <CurrentConnectionsCount>0</CurrentConnectionsCount>
   <MessagesInCount>0</MessagesInCount>
   <MessagesOutCount>0</MessagesOutCount>
   <ASRMessagesReceivedCount>0</ASRMessagesReceivedCount>
   <ASRMessagesSentCount>0</ASRMessagesSentCount>
   <ASASuccessMessagesReceivedCount>0</ASASuccessMessagesReceivedCount>
   <ASASuccessMessagesSentCount>0</ASASuccessMessagesSentCount>
   <ASAFailureMessagesReceivedCount>0</ASAFailureMessagesReceivedCount>
   <ASAFailureMessagesSentCount>0</ASAFailureMessagesSentCount>
   <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
   <RARMessagesSentCount>0</RARMessagesSentCount>
   <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
   <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
   <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
   <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
   <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
   <CCRMessagesSentCount>0</CCRMessagesSentCount>
   <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
```

```
   <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
   <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
   <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
   <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
   <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
   <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
   <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
   <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
   <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
   <ActiveSessionsCount>0</ActiveSessionsCount>
   <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
   <PeerOkayCount>0</PeerOkayCount>
   <PeerDownCount>0</PeerDownCount>
   <PeerSuspectCount>0</PeerSuspectCount>
   <PeerReopenCount>0</PeerReopenCount>
  </Sample>
 </DiameterPcefStats>
</Statistics>
```

Diameter Policy Charging Enforcement Function Peer Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterPcefPeerStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterPcefPeerStats** tag defined in the XSDs.

The following is an example of a request for a single network element, using the Name parameter.

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:35:00Z</EndTime>
    <DiameterPcefNetworkElementStats>
        <Name>Server1</Name>
    </DiameterAfPeerStats>
</QueryOmStats>
```

The following is an example of a request for all network elements in the system. This example returns statistics for each network element.

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2007-01-29T17:30:00Z</StartTime>
    <EndTime>2006-01-29T18:10:00Z</EndTime>
    <DiameterPcefPeerStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DiameterPcefPeerStats** tag defined in the XSDs.

Both Name and NeId are returned along with the statistics and actual recorded times for those statistics. Samples are ordered by policy server, network element, and then by time.

The following is an example XML response to a DiameterPcefPeerStats tag request:

```
<?xml version='1.0' ?>
<Statistics>
 <DiameterPcefPeerStats>
```

```
  <Sample>
   <StartTime>2009-08-19T20:00:00Z</StartTime>
   <EndTime>2009-08-19T20:08:53Z</EndTime>
   <PolicyServer>localps</PolicyServer>
   <IsComplete>true</IsComplete>
   <Name>pcef1</Name>
   <NeId></NeId>
   <NetworkElementType></NetworkElementType>
   <NetworkElementSubType></NetworkElementSubType>
   <ConnectTime></ConnectTime>
   <DisconnectTime></DisconnectTime>
   <ConnectAddress></ConnectAddress>
   <ConnectPort></ConnectPort>
   <MessagesInCount>0</MessagesInCount>
   <MessagesOutCount>0</MessagesOutCount>
   <MessagesErrorInCount>0</MessagesErrorInCount>
   <MessagesErrorOutCount>0</MessagesErrorOutCount>
   <ASRMessagesReceivedCount>0</ASRMessagesReceivedCount>
   <ASRMessagesSentCount>0</ASRMessagesSentCount>
   <ASASuccessMessagesReceivedCount>0</ASASuccessMessagesReceivedCount>
   <ASASuccessMessagesSentCount>0</ASASuccessMessagesSentCount>
   <ASAFailureMessagesReceivedCount>0</ASAFailureMessagesReceivedCount>
   <ASAFailureMessagesSentCount>0</ASAFailureMessagesSentCount>
   <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
   <RARMessagesSentCount>0</RARMessagesSentCount>
   <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
   <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
   <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
   <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
   <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
   <CCRMessagesSentCount>0</CCRMessagesSentCount>
   <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
   <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
   <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
   <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
   <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
   <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
   <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
   <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
   <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
   <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
   <ActiveSessionsCount>0</ActiveSessionsCount>
   <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
  </Sample>
 </DiameterPcefPeerStats>
</Statistics>
```

Diameter Multi-Protocol Routing Agent (MRA) Policy Charging Enforcement Function Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterMraPcefStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterMraPcefStats** tag defined in the XSDs.

The following is an example of this request.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
 <QueryOmStats>
```

```
   <StartTime>2010-04-02T00:01:00</StartTime>
   <EndTime>2011-05-21T23:59:00</EndTime>
     <DiameterMraPcefStats>
    <MRA>mra156</MRA>
   </DiameterMraPcefStats>
 </QueryOmStats>
</XmlInterfaceRequest>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraPcefStats** tag defined in the XSDs. For example:

```
<Statistics>
  <DiameterMraPcefStats>
      <Sample>
      <StartTime>2010-06-04T05:43:15Z</StartTime>
      <EndTime>2010-06-04T05:43:32Z</EndTime>
      <MRA>mra156</MRA>
      <IsComplete>true</IsComplete>
      <CurrentConnectionsCount>0</CurrentConnectionsCount>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <ASRMessagesReceivedCount>0</ASRMessagesReceivedCount>
      <ASRMessagesSentCount>0</ASRMessagesSentCount>
      <ASASuccessMessagesReceivedCount>0</ASASuccessMessagesReceivedCount>
      <ASASuccessMessagesSentCount>0</ASASuccessMessagesSentCount>
      <ASAFailureMessagesReceivedCount>0</ASAFailureMessagesReceivedCount>
      <ASAFailureMessagesSentCount>0</ASAFailureMessagesSentCount>
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
      <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
      <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
      <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
      <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
      <CCRMessagesSentCount>0</CCRMessagesSentCount>
      <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
      <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
      <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
      <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
      <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
      <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
      <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
      <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
      <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
      <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
      <PeerOkayCount>0</PeerOkayCount>
      <PeerDownCount>0</PeerDownCount>
      <PeerSuspectCount>0</PeerSuspectCount>
      <PeerReopenCount>0</PeerReopenCount>
   </Sample>
  </DiameterMraPcefStats>
</Statistics>
```

Diameter Multi-Protocol Routing Agent (MRA) Policy Charging Enforcement Function Peer Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterMraPcefPeerStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterMraPcefPeerStats** tag defined in the XSDs.

The following is an example of this request.

```
<?xml version="1.0" encoding="UTF-8" ?>
 - <XmlInterfaceRequest>
  - <QueryOmStats>
   <StartTime>2010-04-02T00:01:00</StartTime>
   <EndTime>2011-05-21T23:59:00</EndTime>
   - <DiameterMraPcefPeerStats>
    <Name>mra157</Name>
   </DiameterMraPcefPeerStats>
   </QueryOmStats>
  </XmlInterfaceRequest>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraPcefPeerStats** tag defined in the XSDs. For example:

```
- <Statistics>
 - <DiameterMraPcefPeerStats>
  - <Sample>
      <StartTime>2010-05-31T07:56:55Z</StartTime>
      <EndTime>2010-05-31T08:00:00Z</EndTime>
      <MRA>mra156</MRA>
      <IsComplete>true</IsComplete>
      <Name>tekelec2</Name>
      <NeId />
      <NetworkElementType />
      <NetworkElementSubType />
      <ConnectTime>Mon May 31 15:54:23 CST 2010</ConnectTime>
      <DisconnectTime>N/A</DisconnectTime>
      <ConnectAddress>192.168.200.106</ConnectAddress>
      <ConnectPort>56164</ConnectPort>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <MessagesErrorInCount>0</MessagesErrorInCount>
      <MessagesErrorOutCount>0</MessagesErrorOutCount>
      <ASRMessagesReceivedCount>0</ASRMessagesReceivedCount>
      <ASRMessagesSentCount>0</ASRMessagesSentCount>
      <ASASuccessMessagesReceivedCount>0</ASASuccessMessagesReceivedCount>
      <ASASuccessMessagesSentCount>0</ASASuccessMessagesSentCount>
      <ASAFailureMessagesReceivedCount>0</ASAFailureMessagesReceivedCount>
      <ASAFailureMessagesSentCount>0</ASAFailureMessagesSentCount>
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
      <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
      <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
      <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
      <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
      <CCRMessagesSentCount>0</CCRMessagesSentCount>
      <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
      <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
      <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
      <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
      <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
      <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
      <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
      <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
      <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
      <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
```

```
   </Sample>
  </DiameterMraPcefPeerStats>
</Statistics>
```

Diameter Multi-Protocol Routing Agent (MRA) Distributed Routing and Management Application (DRMA) Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterMraDrmaStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterMraDrmaStats** tag defined in the XSDs.

The following is an example of this request.

```
<?xml version="1.0" encoding="UTF-8" ?>
 - <XmlInterfaceRequest>
  - <QueryOmStats>
     <StartTime>2010-04-02T00:01:00</StartTime>
     <EndTime>2011-05-21T23:59:00</EndTime>
   - <DiameterMraDrmaStats>
     <MRA>mra156</MRA>
   </DiameterMraDrmaStats>
   </QueryOmStats>
  </XmlInterfaceRequest>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraDrmaStats** tag defined in the XSDs. For example:

```
- <Statistics>
 - <DiameterMraDrmaStats>
  - <Sample>
      <StartTime>2010-06-04T05:43:15Z</StartTime>
      <EndTime>2010-06-04T05:43:32Z</EndTime>
      <MRA>mra156</MRA>
      <IsComplete>true</IsComplete>
      <CurrentConnectionsCount>0</CurrentConnectionsCount>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <DBRMessagesReceivedCount>0</DBRMessagesReceivedCount>
      <DBRMessagesSentCount>0</DBRMessagesSentCount>
      <DBASuccessMessagesReceivedCount>0</DBASuccessMessagesReceivedCount>
      <DBASuccessMessagesSentCount>0</DBASuccessMessagesSentCount>
      <DBAFailureMessagesReceivedCount>0</DBAFailureMessagesReceivedCount>
      <DBAFailureMessagesSentCount>0</DBAFailureMessagesSentCount>
      <BindingFoundSentCount>0</BindingFoundSentCount>
      <BindingFoundRecvCount>0</BindingFoundRecvCount>
      <BindingNotFoundSentCount>0</BindingNotFoundSentCount>
      <BindingNotFoundRecvCount>0</BindingNotFoundRecvCount>
      <PcrfDownSentCount>0</PcrfDownSentCount>
      <PcrfDownReceivedCount>0</PcrfDownReceivedCount>
      <AllPcrfsDownSentCount>0</AllPcrfsDownSentCount>
      <AllPcrfsDownRecvCount>0</AllPcrfsDownRecvCount>
      <PeerOkayCount>0</PeerOkayCount>
      <PeerDownCount>0</PeerDownCount>
      <PeerSuspectCount>0</PeerSuspectCount>
      <PeerReopenCount>0</PeerReopenCount>
   </Sample>
```

```
  </DiameterMraDrmaStats>
</Statistics>
```

Diameter Multi-Protocol Routing Agent (MRA) Distributed Routing and Management Application (DRMA) over SCTP Statistics

The statistics field can be set to SCTP with the ability to support multiple IP addresses.

The following example show the SCTP response defined in the XSDs for the DiameterMraDrmaPeerStats tag.

Response

The response to this request follows the **Statistics ➤ DiameterMraDrmaPeerStats** tag defined in the XSDs. For example:

```
<Statistics>
  <DiameterMraDrmaPeerStats>
    <Sample>
     <StartTime>2001-12-31T12:00:00</StartTime>
     <EndTime>2001-12-31T12:15:00</EndTime>
     <MRA>MRA</MRA>
     <IsComplete>true</IsComplete>
     <Name>NetworkElementName</Name>
     <ConnectTime>2001-12-31T12:00:00</ConnectTime>
     <DisconnectTime>2001-12-31T12:00:00</DisconnectTime>
     <ConnectType>SCTP</ConnectType>
     <ConnectAddress>10.15.25.110:3868, 10.15.25.119:3868</ConnectAddress>
     <ConnectPort>N/A</ConnectPort>
```

The following example show the TCP response defined in the XSDs for the DiameterMraDrmaPeerStats tag. This diameter connection is set to TCP.

Response

The response to this request follows the **Statistics ➤ DiameterMraDrmaPeerStats** tag defined in the XSDs. For example:

```
<Statistics>
  <DiameterMraDrmaPeerStats>
    <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <MRA>MRA</MRA>
      <IsComplete>true</IsComplete>
      <Name>NetworkElementName</Name>
      <ConnectTime>2001-12-31T12:00:00</ConnectTime>
      <DisconnectTime>2001-12-31T12:00:00</DisconnectTime>
      <ConnectType>N/A</ConnectType>
      <ConnectAddress>10.15.25.110</ConnectAddress>
```

```
      <ConnectPort>3868</ConnectPort>
```

Diameter Multi-Protocol Routing Agent (MRA) Distributed Routing and Management Application Peer Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterMraDrmaPeerStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterMraTdfPeerStats** tag defined in the XSDs.

The following is an example of this request.

```
<?xml version="1.0" encoding="UTF-8" ?>
 - <XmlInterfaceRequest>
  - <QueryOmStats>
    <StartTime>2010-04-02T00:01:00</StartTime>
    <EndTime>2011-05-21T23:59:00</EndTime>
  - <DiameterMraDrmaPeerStats>
    <Name>mra157</Name>
   </DiameterMraDrmaPeerStats>
  </QueryOmStats>
 </XmlInterfaceRequest>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraTdfPeerStats** tag defined in the XSDs. For example:

```
<Statistics>
<DiameterMraTdfPeerStats>
 <Sample>
 <StartTime>2012-05-28T09:30:00Z</StartTime>
 <EndTime>2012-05-28T09:45:00Z</EndTime>
 <MRA>kli-66-mra</MRA>
 <IsComplete>true</IsComplete>
 <Name>kli-63-mpe</Name>
 <NeId/>
 <NetworkElementType/>
 <NetworkElementSubType/>
 <ConnectTime>Fri May 25 05:44:36 EDT 2012</ConnectTime>
 <DisconnectTime>N/A</DisconnectTime>
 <ConnectAddress>10.60.25.64:3868,10.60.25.65:3868</ConnectAddress>
 <ConnectPort>0</ConnectPort>
 <ConnectType/>
 <MessagesInCount>0</MessagesInCount>
 <MessagesOutCount>0</MessagesOutCount>
 <MessagesErrorInCount>0</MessagesErrorInCount>
 <MessagesErrorOutCount>0</MessagesErrorOutCount>
 <TSRMessagesReceivedCount>0</TSRMessagesReceivedCount>
 <TSRMessagesSentCount>0</TSRMessagesSentCount>
 <TSRMessagesTimeoutCount>0</TSRMessagesTimeoutCount>
 <TSASuccessMessagesReceivedCount>0</TSASuccessMessagesReceivedCount>
 <TSASuccessMessagesSentCount>0</TSASuccessMessagesSentCount>
 <TSAFailureMessagesReceivedCount>0</TSAFailureMessagesReceivedCount>
 <TSAFailureMessagesSentCount>0</TSAFailureMessagesSentCount>
 <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
 <RARMessagesSentCount>0</RARMessagesSentCount>
 <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
```

```
<RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
<RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
<RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
<RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
<CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
<CCRMessagesSentCount>0</CCRMessagesSentCount>
<CCRMessagesTimeoutCount>0</CCRMessagesTimeoutCount>
<CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
<CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
<CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
<CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
<CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
<CCRIMessagesSentCount>0</CCRIMessagesSentCount>
<CCRIMessagesTimeoutCount>0</CCRIMessagesTimeoutCount>
<CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
<CCRUMessagesSentCount>0</CCRUMessagesSentCount>
<CCRUMessagesTimeoutCount>0</CCRUMessagesTimeoutCount>
<CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
<CCRTMessagesSentCount>0</CCRTMessagesSentCount>
<CCRTMessagesTimeoutCount>0</CCRTMessagesTimeoutCount>
<CCAISuccessMessagesReceivedCount>0</CCAISuccessMessagesReceivedCount>
<CCAISuccessMessagesSentCount>0</CCAISuccessMessagesSentCount>
<CCAIFailureMessagesReceivedCount>0</CCAIFailureMessagesReceivedCount>
<CCAIFailureMessagesSentCount>0</CCAIFailureMessagesSentCount>
<CCAUSuccessMessagesReceivedCount>0</CCAUSuccessMessagesReceivedCount>
<CCAUSuccessMessagesSentCount>0</CCAUSuccessMessagesSentCount>
<CCAUFailureMessagesReceivedCount>0</CCAUFailureMessagesReceivedCount>
<CCAUFailureMessagesSentCount>0</CCAUFailureMessagesSentCount>
<CCATSuccessMessagesReceivedCount>0</CCATSuccessMessagesReceivedCount>
<CCATSuccessMessagesSentCount>0</CCATSuccessMessagesSentCount>
<CCATFailureMessagesReceivedCount>0</CCATFailureMessagesReceivedCount>
<CCATFailureMessagesSentCount>0</CCATFailureMessagesSentCount>
</Sample>
</DiameterMraTdfPeerStats>
</Statistics>
```

Diameter Multi-Protocol Routing Agent (MRA) Diameter Routing Agent (DRA) Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterMraDraStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterMraDraStats** tag defined in the XSDs.

The following is an example of this request.

```
<?xml version="1.0" encoding="UTF-8" ?>
 - <XmlInterfaceRequest>
  - <QueryOmStats>
    <StartTime>2010-04-02T00:01:00</StartTime>
    <EndTime>2011-05-21T23:59:00</EndTime>
   - <DiameterMraDraStats>
     <MRA>mra156</MRA>
     <MRA>mra106</MRA>
    </DiameterMraDraStats>
   </QueryOmStats>
  </XmlInterfaceRequest>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraDraStats** tag defined in the XSDs. For example:

```
- <Statistics>
 - <DiameterMraDraStats>
  - <Sample>
      <StartTime>2010-06-04T05:43:15Z</StartTime>
      <EndTime>2010-06-04T05:43:32Z</EndTime>
      <MRA>mra106</MRA>
      <IsComplete>true</IsComplete>
      <ActiveBindingsCount>0</ActiveBindingsCount>
      <MaximumActiveBindingsCount>0</MaximumActiveBindingsCount>
   </Sample>
  </DiameterMraDraStats>
</Statistics>
```

Diameter Multi-Protocol Routing Agent (MRA) Application Function (AF) Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterMraAfStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterMraAfStats** tag defined in the XSDs.

The following is an example of this request.

```
<?xml version="1.0" encoding="UTF-8" ?>
 - <XmlInterfaceRequest>
  - <QueryOmStats>
    <StartTime>2010-04-02T00:01:00</StartTime>
    <EndTime>2011-05-21T23:59:00</EndTime>
   - <DiameterMraAfStats>
     <MRA>mra156</MRA>
   </DiameterMraAfStats>
  </QueryOmStats>
 </XmlInterfaceRequest>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraAfStats** tag defined in the XSDs. For example:

```
- <Statistics>
 - <DiameterMraAfStats>
  - <Sample>
      <StartTime>2010-06-04T05:43:15Z</StartTime>
      <EndTime>2010-06-04T05:43:32Z</EndTime>
      <MRA>mra156</MRA>
      <IsComplete>true</IsComplete>
      <PendingConnectionsCount>0</PendingConnectionsCount>
      <CurrentConnectionsCount>0</CurrentConnectionsCount>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <ASRMessagesReceivedCount>0</ASRMessagesReceivedCount>
      <ASRMessagesSentCount>0</ASRMessagesSentCount>
      <ASASuccessMessagesReceivedCount>0</ASASuccessMessagesReceivedCount>
      <ASASuccessMessagesSentCount>0</ASASuccessMessagesSentCount>
      <ASAFailureMessagesReceivedCount>0</ASAFailureMessagesReceivedCount>
      <ASAFailureMessagesSentCount>0</ASAFailureMessagesSentCount>
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
```

```
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
      <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
      <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
      <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
      <STRMessagesReceivedCount>0</STRMessagesReceivedCount>
      <STRMessagesSentCount>0</STRMessagesSentCount>
      <STASuccessMessagesReceivedCount>0</STASuccessMessagesReceivedCount>
      <STASuccessMessagesSentCount>0</STASuccessMessagesSentCount>
      <STAFailureMessagesReceivedCount>0</STAFailureMessagesReceivedCount>
      <STAFailureMessagesSentCount>0</STAFailureMessagesSentCount>
      <AARMessagesReceivedCount>0</AARMessagesReceivedCount>
      <AARMessagesSentCount>0</AARMessagesSentCount>
      <AAASuccessMessagesReceivedCount>0</AAASuccessMessagesReceivedCount>
      <AAASuccessMessagesSentCount>0</AAASuccessMessagesSentCount>
      <AAAFailureMessagesReceivedCount>0</AAAFailureMessagesReceivedCount>
      <AAAFailureMessagesSentCount>0</AAAFailureMessagesSentCount>
      <PeerOkayCount>0</PeerOkayCount>
      <PeerDownCount>0</PeerDownCount>
      <PeerSuspectCount>0</PeerSuspectCount>
      <PeerReopenCount>0</PeerReopenCount>
   </Sample>
  </DiameterMraAfStats>
</Statistics>
```

Diameter Multi-Protocol Routing Agent (MRA) Application Function (AF) Peer Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterMraAfPeerStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterMraAfPeerStats** tag defined in the XSDs.

The following is an example of this request.

```
<?xml version="1.0" encoding="UTF-8" ?>
 - <XmlInterfaceRequest>
  - <QueryOmStats>
    <StartTime>2010-04-02T00:01:00</StartTime>
    <EndTime>2011-05-21T23:59:00</EndTime>
   - <DiameterMraAfPeerStats>
     <Name>mra157</Name>
    </DiameterMraAfPeerStats>
   </QueryOmStats>
  </XmlInterfaceRequest>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraAfPeerStats** tag defined in the XSDs. For example:

```
- <Statistics>
 - <DiameterMraAfPeerStats>
  - <Sample>
      <StartTime>2010-05-31T07:56:55Z</StartTime>
      <EndTime>2010-05-31T08:00:00Z</EndTime>
      <MRA>mra156</MRA>
      <IsComplete>true</IsComplete>
      <Name>tekelec</Name>
      <NeId />
      <NetworkElementType />
```

```
      <NetworkElementSubType />
      <ConnectTime>Mon May 31 15:54:23 CST 2010</ConnectTime>
      <DisconnectTime>N/A</DisconnectTime>
      <ConnectAddress>192.168.200.106</ConnectAddress>
      <ConnectPort>56164</ConnectPort>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <MessagesErrorInCount>0</MessagesErrorInCount>
      <MessagesErrorOutCount>0</MessagesErrorOutCount>
      <ASRMessagesReceivedCount>0</ASRMessagesReceivedCount>
      <ASRMessagesSentCount>0</ASRMessagesSentCount>
      <ASASuccessMessagesReceivedCount>0</ASASuccessMessagesReceivedCount>
      <ASASuccessMessagesSentCount>0</ASASuccessMessagesSentCount>
      <ASAFailureMessagesReceivedCount>0</ASAFailureMessagesReceivedCount>
      <ASAFailureMessagesSentCount>0</ASAFailureMessagesSentCount>
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
      <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
      <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
      <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
      <STRMessagesReceivedCount>0</STRMessagesReceivedCount>
      <STRMessagesSentCount>0</STRMessagesSentCount>
      <STASuccessMessagesReceivedCount>0</STASuccessMessagesReceivedCount>
      <STASuccessMessagesSentCount>0</STASuccessMessagesSentCount>
      <STAFailureMessagesReceivedCount>0</STAFailureMessagesReceivedCount>
      <STAFailureMessagesSentCount>0</STAFailureMessagesSentCount>
      <AARMessagesReceivedCount>0</AARMessagesReceivedCount>
      <AARMessagesSentCount>0</AARMessagesSentCount>
      <AAASuccessMessagesReceivedCount>0</AAASuccessMessagesReceivedCount>
      <AAASuccessMessagesSentCount>0</AAASuccessMessagesSentCount>
      <AAAFailureMessagesReceivedCount>0</AAAFailureMessagesReceivedCount>
      <AAAFailureMessagesSentCount>0</AAAFailureMessagesSentCount>
   </Sample>
  </DiameterMraAfPeerStats>
</Statistics>
```

Diameter Multi-Protocol Routing Agent (MRA) Bearer Binding and Event Reporting Function (Bberf) Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterMraBberfStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterMraBberfStats** tag defined in the XSDs.

The following is an example of this request.

```
<?xml version="1.0" encoding="UTF-8" ?>
 - <XmlInterfaceRequest>
  - <QueryOmStats>
    <StartTime>2010-04-02T00:01:00</StartTime>
    <EndTime>2011-05-21T23:59:00</EndTime>
   - <DiameterMraBberfStats>
     <MRA>mra156</MRA>
    </DiameterMraBberfStats>
   </QueryOmStats>
  </XmlInterfaceRequest>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraBberfStats** tag defined in the XSDs. For example:

```
- <Statistics>
 - <DiameterMraBberfStats>
  - <Sample>
      <StartTime>2010-06-04T05:43:15Z</StartTime>
      <EndTime>2010-06-04T05:43:32Z</EndTime>
      <MRA>mra156</MRA>
      <IsComplete>true</IsComplete>
      <CurrentConnectionsCount>0</CurrentConnectionsCount>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <MessagesErrorInCount>0</MessagesErrorInCount>
      <MessagesErrorOutCount>0</MessagesErrorOutCount>
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
      <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
      <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
      <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
      <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
      <CCRMessagesSentCount>0</CCRMessagesSentCount>
      <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
      <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
      <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
      <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
      <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
      <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
      <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
      <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
      <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
      <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
      <PeerOkayCount>0</PeerOkayCount>
      <PeerDownCount>0</PeerDownCount>
      <PeerSuspectCount>0</PeerSuspectCount>
      <PeerReopenCount>0</PeerReopenCount>
   </Sample>
  </DiameterMraBberfStats>
</Statistics>
```

Diameter Multi-Protocol Routing Agent (MRA) Traffic Detection Function (TDF) Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterMraTdfStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterMraTdfStats** tag defined in the XSDs.

The following is an example of this request.

```
<?xml version="1.0" encoding="UTF-8" ?>
-<XmlInterfaceRequest>
  -<QueryOmStats> <StartTime>2010-04-02T00:01:00</StartTime>
<EndTime>2011-05-21T23:59:00</EndTime>
    -<DiameterMraTdfStats>
      <MRA>mra156</MRA>
     </DiameterMraTdfStats>
```

```
   </QueryOmStats>
</XmlInterfaceRequest>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraTdfStats** tag defined in the XSDs. For example:

```
<Statistics>
 <DiameterMraTdfStats>
 <Sample>
   <StartTime>2012-05-28T09:30:00Z</StartTime>
   <EndTime>2012-05-28T09:45:00Z</EndTime>
   <MRA>kli-66-mra</MRA>
   <IsComplete>true</IsComplete>
   <CurrentConnectionsCount>1</CurrentConnectionsCount>
   <MessagesInCount>0</MessagesInCount>
   <MessagesOutCount>0</MessagesOutCount>
   <TSRMessagesReceivedCount>0</TSRMessagesReceivedCount>
   <TSRMessagesSentCount>0</TSRMessagesSentCount>
   <TSRMessagesTimeoutCount>0</TSRMessagesTimeoutCount>
   <TSASuccessMessagesReceivedCount>0</TSASuccessMessagesReceivedCount>
   <TSASuccessMessagesSentCount>0</TSASuccessMessagesSentCount>
   <TSAFailureMessagesReceivedCount>0</TSAFailureMessagesReceivedCount>
   <TSAFailureMessagesSentCount>0</TSAFailureMessagesSentCount>
   <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
   <RARMessagesSentCount>0</RARMessagesSentCount>
   <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
   <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
   <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
   <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
   <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
   <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
   <CCRMessagesSentCount>0</CCRMessagesSentCount>
   <CCRMessagesTimeoutCount>0</CCRMessagesTimeoutCount>
   <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
   <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
   <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
   <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
   <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
   <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
   <CCRIMessagesTimeoutCount>0</CCRIMessagesTimeoutCount>
   <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
   <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
   <CCRUMessagesTimeoutCount>0</CCRUMessagesTimeoutCount>
   <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
   <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
   <CCRTMessagesTimeoutCount>0</CCRTMessagesTimeoutCount>
   <CCAISuccessMessagesReceivedCount>0</CCAISuccessMessagesReceivedCount>
   <CCAISuccessMessagesSentCount>0</CCAISuccessMessagesSentCount>
   <CCAIFailureMessagesReceivedCount>0</CCAIFailureMessagesReceivedCount>
   <CCAIFailureMessagesSentCount>0</CCAIFailureMessagesSentCount>
   <CCAUSuccessMessagesReceivedCount>0</CCAUSuccessMessagesReceivedCount>
   <CCAUSuccessMessagesSentCount>0</CCAUSuccessMessagesSentCount>
   <CCAUFailureMessagesReceivedCount>0</CCAUFailureMessagesReceivedCount>
   <CCAUFailureMessagesSentCount>0</CCAUFailureMessagesSentCount>
   <CCATSuccessMessagesReceivedCount>0</CCATSuccessMessagesReceivedCount>
   <CCATSuccessMessagesSentCount>0</CCATSuccessMessagesSentCount>
   <CCATFailureMessagesReceivedCount>0</CCATFailureMessagesReceivedCount>
   <CCATFailureMessagesSentCount>0</CCATFailureMessagesSentCount>
   <PeerOkayCount>1</PeerOkayCount>
   <PeerDownCount>0</PeerDownCount>
   <PeerSuspectCount>0</PeerSuspectCount>
   <PeerReopenCount>0</PeerReopenCount>
```

```
  </Sample>
  </DiameterMraTdfStats>
</Statistics>
```

Diameter Multi-Protocol Routing Agent (MRA) Traffic Detection Function (TDF) Peer Statistics

The following examples show the request and response that are defined in the XSDs for the DiameterMraTdfPeerStats tag.

Request

This request follows the **QueryOmStats ➤ DiameterMraTdfPeerStats** tag defined in the XSDs.

The following is an example of this request.

```
<<?xml version="1.0" encoding="UTF-8" ?>
-<XmlInterfaceRequest>
 -<QueryOmStats> <StartTime>2010-04-02T00:01:00</StartTime>
  <EndTime>2011-05-21T23:59:00</EndTime>
 -<DiameterMraTdfPeerStats>
     <Name>mra157</Name>
     </DiameterMraTdfPeerStats>
 </QueryOmStats>
</XmlInterfaceRequest>
```

Response

The response to this request follows the **Statistics ➤ DiameterMraTdfPeerStats** tag defined in the XSDs. For example:

```
<Statistics>
  <DiameterMraTdfStats>
    <Sample>
      <StartTime>2012-05-28T09:30:00Z</StartTime>
      <EndTime>2012-05-28T09:45:00Z</EndTime>
      <MRA>kli-66-mra</MRA>
      <IsComplete>true</IsComplete>
      <IsComplete>true</IsComplete>
      <CurrentConnectionsCount>1</CurrentConnectionsCount>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <TSRMessagesReceivedCount>0</TSRMessagesReceivedCount>
      <TSRMessagesSentCount>0</TSRMessagesSentCount>
      <TSRMessagesTimeoutCount>0</TSRMessagesTimeoutCount>
      <TSASuccessMessagesReceivedCount>0</TSASuccessMessagesReceivedCount>
      <TSASuccessMessagesSentCount>0</TSASuccessMessagesSentCount>
      <TSAFailureMessagesReceivedCount>0</TSAFailureMessagesReceivedCount>
      <TSAFailureMessagesSentCount>0</TSAFailureMessagesSentCount>
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
      <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
      <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
      <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
      <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
      <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
      <CCRMessagesSentCount>0</CCRMessagesSentCount>
      <CCRMessagesTimeoutCount>0</CCRMessagesTimeoutCount>
      <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
      <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
```

```
      <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
      <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
      <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
      <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
      <CCRIMessagesTimeoutCount>0</CCRIMessagesTimeoutCount>
      <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
      <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
      <CCRUMessagesTimeoutCount>0</CCRUMessagesTimeoutCount>
      <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
      <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
      <CCRTMessagesTimeoutCount>0</CCRTMessagesTimeoutCount>
      <CCAISuccessMessagesReceivedCount>0</CCAISuccessMessagesReceivedCount>
      <CCAISuccessMessagesSentCount>0</CCAISuccessMessagesSentCount>
      <CCAIFailureMessagesReceivedCount>0</CCAIFailureMessagesReceivedCount>
      <CCAIFailureMessagesSentCount>0</CCAIFailureMessagesSentCount>
      <CCAUSuccessMessagesReceivedCount>0</CCAUSuccessMessagesReceivedCount>
      <CCAUSuccessMessagesSentCount>0</CCAUSuccessMessagesSentCount>
      <CCAUFailureMessagesReceivedCount>0</CCAUFailureMessagesReceivedCount>
      <CCAUFailureMessagesSentCount>0</CCAUFailureMessagesSentCount>
      <CCATSuccessMessagesReceivedCount>0</CCATSuccessMessagesReceivedCount>
      <CCATSuccessMessagesSentCount>0</CCATSuccessMessagesSentCount>
      <CCATFailureMessagesReceivedCount>0</CCATFailureMessagesReceivedCount>
      <CCATFailureMessagesSentCount>0</CCATFailureMessagesSentCount>
      <PeerOkayCount>1</PeerOkayCount>
      <PeerDownCount>0</PeerDownCount>
      <PeerSuspectCount>0</PeerSuspectCount>
      <PeerReopenCount>0</PeerReopenCount>
    </Sample>
  </DiameterMraTdfStats>
</Statistics>
```

Diameter Multi-Protocol Routing Agent (MRA) Traffic Detection Function (TDF) over SCTP Statistics

The statistics field can be set to SCTP with the ability to support multiple IP addresses.

The following examples show the SCTP response defined in the XSDs for the DiameterMraDrmaPeerStats tag.

Response

The response to this request follows the **Statistics ➤ DiameterMraTdfPeerStats** tag defined in the XSDs. For example:

The following is an example of this request.

```
<Statistics> <DiameterMraTdfPeerStats>
   <Sample> <StartTime>2001-12-31T12:00:00</StartTime>
   <EndTime>2001-12-31T12:15:00</EndTime> <MRA>MRA</MRA>
   <Name>NetworkElementName</Name>
   <ConnectTime>2001-12-31T12:00:00</ConnectTime>
   <DisconnectTime>2001-12-31T12:00:00</DisconnectTime>
   <ConnectType>SCTP</ConnectType>
   <ConnectAddress>10.15.25.110:3868, 10.15.25.119:3868</ConnectAddress>
   <ConnectPort>N/A</ConnectPort>
```

The following example show the TCP response defined in the XSDs for the DiameterMraTdfPeerStats tag. This diameter connection is set to TCP.

Response

The response to this request follows the **Statistics ➤ DiameterMraTdfPeerStats** tag defined in the XSDs. For example:

```
<Statistics>
   <DiameterMraTdfPeerStats> <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <MRA>MRA</MRA>
      <IsComplete>true</IsComplete>
         <Name>NetworkElementName</Name>
         <ConnectTime>2001-12-31T12:00:00</ConnectTime>
         <DisconnectTime>2001-12-31T12:00:00</DisconnectTime>
         <ConnectType>TCP</ConnectType>
         <ConnectAddress>10.15.25.110</ConnectAddress>
         <ConnectPort>3868</ConnectPort>
```

KPI Statistics

The following examples show the request and response that are defined in the XSDs for the KpiStats tag.

Request

This request follows the **QueryOmStats ➤ KpiStats** tag defined in the XSDs.

The following is an example of this request.

```
<?xml version="1.0" encoding="UTF-8" ?>
  - <QueryOmStats>
    <StartTime>2001-12-31T12:00:00</StartTime>
    <EndTime>2001-12-31T13:00:00</EndTime>
  - <KpiStats>
    <PolicyServer>mpe85</PolicyServer>
    <MRA>mra36</MRA>
    </KpiStats>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ KpiStats** tag defined in the XSDs. For example:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <Statistics>
 - <KpiStats>
  - <Sample>
  <StartTime>2001-12-31T12:00:00Z</StartTime>
  <EndTime>2001-12-31T12:15:00Z</EndTime>
  <PolicyServer>mpe85</PolicyServer>
  <IsComplete>true</IsComplete>
  <TransactionStartCount>0</TransactionStartCount>
  <TransactionEndCount>0</TransactionEndCount>
  <MessagesInCount>0</MessagesInCount>
  <MessagesOutCount>0</MessagesOutCount>
  <CurrentTransactionsPerSecond>0</CurrentTransactionsPerSecond>
  <MaxTransactionsPerSecond>0</MaxTransactionsPerSecond>
  <CurrentTPSPercentageOfCapacity>0</CurrentTPSPercentageOfCapacity>
  <MaxTPSPercentageOfCapacity>0</MaxTPSPercentageOfCapacity>
  <CurrentLTETransactionsPerSecond>0</CurrentLTETransactionsPerSecond>
```

```
 <CurrentEHRPDTransactionsPerSecond>0</CurrentEHRPDTransactionsPerSecond>
 <CurrentRXTransactionsPerSecond>0</CurrentRXTransactionsPerSecond>
 <LoadSheddingStatus>0</LoadSheddingStatus>
 <LoadSheddingEfficiency>0</LoadSheddingEfficiency>
 <LoadSheddingDistressCount>0</LoadSheddingDistressCount>
 <CurrentMRABindingCount>0</CurrentMRABindingCount>
 <MaxMRABindingCount>0</MaxMRABindingCount>
 <TotalMRABindingCount>0</TotalMRABindingCount>
 <CurrentSessionCount>0</CurrentSessionCount>
 <MaxSessionCount>0</MaxSessionCount>
 <CurrentPDNConnectionCount>0</CurrentPDNConnectionCount>
 <MaxPDNConnectionCount>0</MaxPDNConnectionCount>
 <CurrentPDNConnectionPercentageOfCapacity>0</CurrentPDN
    ConnectionPercentageOfCapacity>
 <MaxPDNConnectionPercentageOfCapacity>0</MaxPDNConnectionPercentageOfCapacity>
 <CurrentSessionPercentageOfCapacity>0</CurrentSessionPercentageOfCapacity>
 <MaxSessionPercentageOfCapacity>0</MaxSessionPercentageOfCapacity>
 <CurrentMPEConnectionCount>0</CurrentMPEConnectionCount>
 <ConfiguredMPEConnectionCount>0</ConfiguredMPEConnectionCount>
 <CurrentDRMAConnectionCount>0</CurrentDRMAConnectionCount>
 <ConfiguredDRMAConnectionCount>0</ConfiguredDRMAConnectionCount>
 <CurrentConnectedNECount>0</CurrentConnectedNECount>
 <ConfiguredNECount>0</ConfiguredNECount>
 <CurrentMRAConnectionCount>0</CurrentMRAConnectionCount>
 <ConfiguredMRAConnectionCount>0</ConfiguredMRAConnectionCount>
 <CurrentSPRConnectionCount>0</CurrentSPRConnectionCount>
 <ConfiguredSPRConnectionCount>0</ConfiguredSPRConnectionCount>
 <PrimaryCPUUtilizationPercentage>0</PrimaryCPUUtilizationPercentage>
 <PrimaryMemoryUtilizationPercentage>0</PrimaryMemoryUtilizationPercentage>
 <PrimaryDiskUtilizationPercentage>0</PrimaryDiskUtilizationPercentage>
 <PrimaryServerFailureCount>0</PrimaryServerFailureCount>
 <PrimaryUpTimeMillis>0</PrimaryUpTimeMillis>
 <PrimaryServerStatus>0</PrimaryServerStatus>
 <SecondaryCPUUtilizationPercentage>0</SecondaryCPUUtilizationPercentage>
 <SecondaryMemoryUtilizationPercentage>0</SecondaryMemoryUtilizationPercentage>
 <SecondaryDiskUtilizationPercentage>0</SecondaryDiskUtilizationPercentage>
 <SecondaryServerFailureCount>0</SecondaryServerFailureCount>
 <SecondaryUpTimeMillis>0</SecondaryUpTimeMillis>
 <SecondaryServerStatus>0</SecondaryServerStatus>
</Sample>
- <Sample>
 <StartTime>2001-12-31T12:00:00Z</StartTime>
 <EndTime>2001-12-31T12:15:00Z</EndTime>
 <MRA>mra36</MRA>
 <IsComplete>true</IsComplete>
 <TransactionStartCount>0</TransactionStartCount>
 <TransactionEndCount>0</TransactionEndCount>
 <MessagesInCount>0</MessagesInCount>
 <MessagesOutCount>0</MessagesOutCount>
 <CurrentTransactionsPerSecond>0</CurrentTransactionsPerSecond>
 <MaxTransactionsPerSecond>0</MaxTransactionsPerSecond>
 <CurrentTPSPercentageOfCapacity>0</CurrentTPSPercentageOfCapacity>
 <MaxTPSPercentageOfCapacity>0</MaxTPSPercentageOfCapacity>
 <CurrentLTETransactionsPerSecond>0</CurrentLTETransactionsPerSecond>
 <CurrentEHRPDTransactionsPerSecond>0</CurrentEHRPDTransactionsPerSecond>
 <CurrentRXTransactionsPerSecond>0</CurrentRXTransactionsPerSecond>
 <LoadSheddingStatus>0</LoadSheddingStatus>
 <LoadSheddingEfficiency>0</LoadSheddingEfficiency>
 <LoadSheddingDistressCount>0</LoadSheddingDistressCount>
 <CurrentMRABindingCount>0</CurrentMRABindingCount>
 <MaxMRABindingCount>0</MaxMRABindingCount>
 <TotalMRABindingCount>0</TotalMRABindingCount>
 <CurrentSessionCount>0</CurrentSessionCount>
 <MaxSessionCount>0</MaxSessionCount>
 <CurrentPDNConnectionCount>0</CurrentPDNConnectionCount>
```

```
  <MaxPDNConnectionCount>0</MaxPDNConnectionCount>
  <CurrentPDNConnectionPercentageOfCapacity>0</CurrentPDN
    ConnectionPercentageOfCapacity>
  <MaxPDNConnectionPercentageOfCapacity>0</MaxPDNConnectionPercentageOfCapacity>
  <CurrentSessionPercentageOfCapacity>0</CurrentSessionPercentageOfCapacity>
  <MaxSessionPercentageOfCapacity>0</MaxSessionPercentageOfCapacity>
  <CurrentMPEConnectionCount>0</CurrentMPEConnectionCount>
  <ConfiguredMPEConnectionCount>0</ConfiguredMPEConnectionCount>
  <CurrentDRMAConnectionCount>0</CurrentDRMAConnectionCount>
  <ConfiguredDRMAConnectionCount>0</ConfiguredDRMAConnectionCount>
  <CurrentConnectedNECount>0</CurrentConnectedNECount>
  <ConfiguredNECount>0</ConfiguredNECount>
  <CurrentMRAConnectionCount>0</CurrentMRAConnectionCount>
  <ConfiguredMRAConnectionCount>0</ConfiguredMRAConnectionCount>
  <CurrentSPRConnectionCount>0</CurrentSPRConnectionCount>
  <ConfiguredSPRConnectionCount>0</ConfiguredSPRConnectionCount>
  <PrimaryCPUUtilizationPercentage>0</PrimaryCPUUtilizationPercentage>
  <PrimaryMemoryUtilizationPercentage>0</PrimaryMemoryUtilizationPercentage>
  <PrimaryDiskUtilizationPercentage>0</PrimaryDiskUtilizationPercentage>
  <PrimaryServerFailureCount>0</PrimaryServerFailureCount>
  <PrimaryUpTimeMillis>0</PrimaryUpTimeMillis>
  <PrimaryServerStatus>0</PrimaryServerStatus>
  <SecondaryCPUUtilizationPercentage>0</SecondaryCPUUtilizationPercentage>
  <SecondaryMemoryUtilizationPercentage>0</SecondaryMemoryUtilizationPercentage>
  <SecondaryDiskUtilizationPercentage>0</SecondaryDiskUtilizationPercentage>
  <SecondaryServerFailureCount>0</SecondaryServerFailureCount>
  <SecondaryUpTimeMillis>0</SecondaryUpTimeMillis>
  <SecondaryServerStatus>0</SecondaryServerStatus>
 </Sample>
</KpiStats>
</Statistics>
```

PCMM CMTS Statistics

The following examples show the request and response that are defined in the XSDs for the PcmmCmtsStats tag.

Request

This request follows the **QueryOmStats ➤ PcmmCmtsStats** tag defined in the XSDs.

The following is an example of a request for all policy servers in the system.

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:40:00Z</EndTime>
    <PcmmCmtsStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ PcmmCmtsStats** tag defined in the XSDs.

SessionCount — The number of session requests received.

SessionSuccessCount — The number of session requests successfully created.

SessionFailCount — The number of session request failures. For example, this could be due to a policy denying a request.

SessionProtocolFailCount — The number of session failures due to an invalid message or parameter. This count is incremented whenever the MPE determines that an incoming message from the AM has an invalid message and as a result the message has to be dropped by the MPE.

SessionPolicyFailCount — The number of session requests that trigger a policy. This count is maintained in the MPE, one per policy. When the condition of a policy triggers, the count for that policy is incremented. The value displayed is a total trigger count, (that is, the sum of this value for all the policies). For example, if the definition of a policy is defined as, "when the device usage is greater than 80% of capacity, reject message". When the MPE executes this policy in response to a request, if the policy triggers (that is, if the MPE determines that the device usage is > 80%), the trigger count for that policy is incremented.

The following is an example XML response to a PcmmCmtsStats tag request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
    <PcmmCmtsStats>
        <Sample>
            <StartTime>2006-10-26T14:30:22Z</StartTime>
            <EndTime>2006-10-26T14:35:11Z</EndTime>
            <PolicyServer>Atlanta105</PolicyServer>
            <SessionCount>100</SessionCount>
            <SessionSuccessCount>97</SessionSuccessCount>
            <SessionFailCount>3</SessionFailCount>
            <SessionProtocolFailCount>1</SessionProtocolFailCount>
            <SessionPolicyFailCount>2</SessionPolicyFailCount>
        </Sample>
    </PcmmCmtsStats>
</Statistics>
```

PCMM Network Element Statistics

The following examples show the request and response that are defined in the XSDs for the PcmmNetworkElementStats tag.

Request

This request follows the **QueryOmStats ➤ PcmmNetworkElementStats** tag defined in the XSDs.

The following is an example of a request for a single PCMM network element using the Name parameter:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:35:00Z</EndTime>
    <PcmmNetworkElementStats>
        <Name>Router1</Name>
    </PcmmNetworkElementStats>
</QueryOmStats>
```

The following is an example of a request for multiple PCMM network elements using the Name(s) and NeId(s) parameters. This example returns statistics for three different network elements:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
```

```
    <StartTime>2006-10-12T11:15:00Z</StartTime>
    <EndTime>2006-10-12T11:25:00Z</EndTime>
    <PcmmNetworkElementStats>
        <Name>Router1</Name>
        <Name>Router2</Name>
        <Name>Router3</Name>
    </PcmmNetworkElementStats>
</QueryOmStats>
```

The following is an example of a request for all PCMM network elements in the system. This example returns statistics for each PCMM network element:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2006-10-12T11:15:00Z</StartTime>
    <EndTime>2006-10-12T11:25:00Z</EndTime>
    <PcmmNetworkElementStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ PcmmNetworkElementStats** tag defined in the XSDs.

Both Name and NeId are returned along with the statistics and actual recorded times for those statistics. Sample groups are ordered by policy server, network element, available interface, and then by time.

- Name — Unique name identifying the network element for the following statistics.
  - NeId — Optional identifier field for the network element.
  - InterfaceName — Identifying field for the network element's Interface.
  - SessionCount — Current active sessions for that network element or Interface. This is a non-cumulative value and is displayed as an absolute.
  - SessionSuccessCount — Successful sessions.
  - SessionFailCount — Session failures.
  - Capacity — The currently defined, maximum capacity for this network element or Interface. This is a static absolute value defined in the CMP for that object.

The following is an example XML response to a multiple network element request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
    <PcmmNetworkElementStats>
        <Sample>
            <StartTime>2006-10-12T11:18:30Z</StartTime>
            <EndTime>2006-10-12T11:19:20Z</EndTime>
            <PolicyServer>Atlanta105</PolicyServer>
            <Name>Router1</Name>
            <NeId>12345</NeId>
            <SessionCount>3</SessionCount>
            <SessionSuccessCount>3</SessionSuccessCount>
            <SessionFailCount>0</SessionFailCount>
            <Capacity>50000000</Capacity>
        </Sample>
        <Sample>
            <StartTime>2006-10-12T11:19:20Z</StartTime>
            <EndTime>2006-10-12T11:20:10Z</EndTime>
            <PolicyServer>Atlanta105</PolicyServer>
            <Name>Router1</Name>
            <NeId>12345</NeId>
```

```
            <SessionCount>0</SessionCount>
            <SessionSuccessCount>0</SessionSuccessCount>
            <SessionFailCount>0</SessionFailCount>
            <Capacity>50000000</Capacity>
        </Sample>
    </PcmmNetworkElementStats>
</Statistics>
```

DQoS CMTS Statistics

The following examples show the request and response that are defined in the XSDs for the DqosCmtsStats tag.

Request

This request follows the **QueryOmStats ➤ DqosCmtsStats** tag defined in the XSDs.

The following is an example of a request for all policy servers in the system.

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:40:00Z</EndTime>
    <DqosCmtsStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DqosCmtsStats** tag defined in the XSDs.

SessionCount — The number of session requests received.

SessionSuccessCount — The number of session requests successfully created.

SessionFailCount — The number of session request failures. For example, this could be due to a policy denying a request.

SessionProtocolFailCount — The number of session failures due to an invalid message or parameter. This count is incremented whenever the MPE determines that an incoming message from the AM has an invalid message and as a result, the message has to be dropped by the MPE.

SessionPolicyFailCount — The number of session requests that trigger a policy. This count is maintained in the MPE, one per policy. When the condition of a policy triggers, the count for that policy is incremented. The value displayed is a total trigger count, (that is, the sum of this value for all the policies). For example, if the definition of a policy is defined as, “when the device usage is greater than 80% of capacity, reject message”. When the MPE executes this policy in response to a request, if the policy triggers (that is, if the MPE determines that the device usage is > 80%), the trigger count for that policy is incremented.

The following is an example XML response to a DqosCmtsStats tag request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
    <DqosCmtsStats>
        <Sample>
            <StartTime>2006-10-26T14:30:22Z</StartTime>
            <EndTime>2006-10-26T14:35:11Z</EndTime>
            <PolicyServer>Atlanta105</PolicyServer>
```

```
            <SessionCount>100</SessionCount>
            <SessionSuccessCount>97</SessionSuccessCount>
            <SessionFailCount>3</SessionFailCount>
            <SessionProtocolFailCount>1</SessionProtocolFailCount>
            <SessionPolicyFailCount>2</SessionPolicyFailCount>
        </Sample>
    </DqosCmtsStats>
</Statistics>
```

DQos Network Element Statistics

The following examples show the request and response that are defined in the XSDs for the DqosNetworkElementStats tag.

Request

This request follows the **QueryOmStats ➤ DqosNetworkElementStats** tag defined in the XSDs.

The following is an example of a request for a single DQoS network element using the Name parameter:

```
<?xml version="1.0" encoding="UTF-8" ?>
<QueryOmStats>
    <StartTime>2006-10-26T14:30:00Z</StartTime>
    <EndTime>2006-10-26T14:35:00Z</EndTime>
    <DqosNetworkElementStats>
        <Name>Router1</Name>
    </DqosNetworkElementStats>
</QueryOmStats>
```

The following is an example of a request for all DQoS network elements in the system. This example returns statistics for each DQoS network element:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2006-10-12T11:15:00Z</StartTime>
    <EndTime>2006-10-12T11:25:00Z</EndTime>
    <DqosNetworkElementStats/>
</QueryOmStats>
```

Response

The response to this request follows the **Statistics ➤ DqosNetworkElementStats** tag defined in the XSDs.

Both Name and NeId are returned along with the statistics and actual recorded times for those statistics. Sample groups are ordered by policy server, network element, available interface, and then by time.

- Name — Unique name identifying the network element for the following statistics.
  - NeId — Optional identifier field for the network element.
  - InterfaceName — Identifying field for the network element's Interface.
  - SessionCount — Current active sessions for that network element or Interface. This is a non-cumulative value and is displayed as an absolute.
  - SessionSuccessCount — Successful sessions.
  - SessionFailCount — Session failures.

- Capacity — The currently defined, maximum capacity for this network element or Interface. This is a static absolute value defined in the CMP for that object.

The following is an example XML response to a multiple network element request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
    <DqosNetworkElementStats>
        <Sample>
            <StartTime>2006-10-12T11:18:30Z</StartTime>
            <EndTime>2006-10-12T11:19:20Z</EndTime>
            <PolicyServer>Atlanta105</PolicyServer>
            <Name>Router1</Name>
            <NeId>12345</NeId>
            <SessionCount>3</SessionCount>
            <SessionSuccessCount>3</SessionSuccessCount>
            <SessionFailCount>0</SessionFailCount>
            <Capacity>50000000</Capacity>
        </Sample>
        <Sample>
            <StartTime>2006-10-12T11:19:20Z</StartTime>
            <EndTime>2006-10-12T11:20:10Z</EndTime>
            <PolicyServer>Atlanta105</PolicyServer>
            <Name>Router1</Name>
            <NeId>12345</NeId>
            <SessionCount>0</SessionCount>
            <SessionSuccessCount>0</SessionSuccessCount>
            <SessionFailCount>0</SessionFailCount>
            <Capacity>50000000</Capacity>
        </Sample>
    </DqosNetworkElementStats>
</Statistics>
```

Protocol Error Statistics

The following examples show the request and response that are defined in the XSDs for the ProtocolErrorStats tag.

Request

This request follows the **QueryOmStats ➤ ProtocolErrorStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<XmlInterfaceRequest>
  <QueryOmStats DeltaCount="false">
    <StartTime>2010-04-02T00:01:00</StartTime>
    <EndTime>2011-05-31T23:59:00</EndTime>
  <ProtocolErrorStats>
    <PolicyServer>rc178</PolicyServer>
  </ProtocolErrorStats>
  </QueryOmStats>
</XmlInterfaceRequest>
```

The response to this request follows the **Statistics ➤ ProtocolErrorStats** tag defined in the XSDs.

Individual statistics are defined as follows:

- DiameterCommandUnsupported - The request contained a command code that the receiver did not recognize or support.
- DiameterUnableToDeliver - This error is given when the diameter node cannot deliver the message to the destination, either because no host within the realm that supports the required application was available to process the request, or because Destination-Host AVP was given without the associated Destination-Realm AVP.
- DiameterRealmNotServed - The intended realm of the request is not recognized.
- DiameterTooBusy - The diameter node has too much traffic. When returned, a diameter node attempts to send the message to an alternate peer.
- DiameterLoopDetected - An agent detected a loop while trying to get the message to the intended recipient. The message may be sent to an alternate peer if available.
- DiameterRedirectIndication - A redirect agent has determined that the request could not be satisfied locally and the initiator of the request should direct the request directly to the server, whose contact information has been added to the response. When set, the Redirect-Host AVP must be present.
- DiameterApplicationUnsupported - A request was sent for an application that is not supported.
- DiameterInvalidHdrBits - A request was received whose bits in the diameter header were either set to an invalid combination, or to a value that is inconsistent with the command code's definition.
- DiameterInvalidAvpBits - A request was received that included an AVP whose flag bits are set to an unrecognized value, or that is inconsistent with the AVP's definition.
- DiameterUnknownPeer - A CER was received from an unknown peer.
- DiameterAuthenticationRejected - The authentication process for the user failed, most likely due to an invalid password used by the user. The user is then prompted for a new password.
- DiameterOutOfSpace - A diameter node received the accounting request but was unable to commit it to stable storage due to a temporary lack of space.
- ElectionLost - The peer has lost the election process and has disconnected the transport connection.
- DiameterEndUserServiceDenied - The credit-control server denied the service request due to service restrictions. If the CCR contained used-service-units, they are deducted, if possible.
- DiameterCreditControlNotApplicable - The credit-control server determined that the service can be granted to the end user, but that no further credit control is needed for the service (for example, the service is free of charge).
- DiameterCreditLimitReached - The credit-control server denied the service request because the end user's account could not cover the requested service. If the CCR contained used-service-units, they are deducted, if possible.
- DiameterAvpUnsupported - The peer received a message that contained an AVP that is not recognized or supported and was marked with the Mandatory bit.
- DiameterUnknownSessionId - The request contained an unknown Session-Id.
- DiameterAuthorizationRejected - A request was received for which the user could not be authorized. This error could occur if the requested service is not permitted to the user.
- DiameterInvalidAvpValue - The request contained an AVP with an invalid value in its data portion.
- DiameterMissingAvp - The request did not contain an AVP that is required by the command code definition. If this value is sent in the Result-Code AVP, a Failed-AVP AVP is included in the message.
- DiameterResourcesExceeded - A request was received that cannot be authorized because the user has already expended its allowed resources.
- DiameterContradictingAvps - The Home Diameter server has detected AVPs in the request that contradict each other, and is not willing to provide service to the user.
- DiameterAvpNotAllowed - A message was received with an AVP that cannot be present.
- DiameterAvpOccursTooManyTimes - A message was received that included an AVP that appeared more often than permitted.

- DiameterNoCommonApplication - This error is returned when a CER message is received, and there are no common applications supported between the peers.
- DiameterUnsupportedVersion - This error is returned when a request was received with an unsupported version number.
- DiameterUnableToComply - This error is returned when a request is rejected for unspecified reasons.
- DiameterInvalidBitInHeader - This error is returned when an unrecognized bit in the diameter header is set to one (1).
- DiameterInvalidAvpLength - The request contained an AVP with an invalid length.
- DiameterInvalidMessageLength - This error is returned when a request is received with an invalid message length.
- DiameterInvalidAvpBitCombo - The request contained an AVP with an invalid AVP Flags value.
- DiameterNoCommonSecurity - This error is returned when a CER message is received, but there are no common security mechanisms supported between the peers.
- DiameterUserUnknown - The specified end user is unknown in the credit-control server.
- DiameterRatingFailed - This error code is used to inform the credit-control client that the credit-control server cannot rate the service request due to insufficient rating input, an incorrect AVP combination, or an AVP or AVP value that is not recognized or supported in the rating.
- DiameterErrorInitialParameters - The initial parameters contain an error.
- RadiusSessionContyextRemoved - The residual session context has been removed.
- RadiusInvalidEapPacket - An invalid EAP Packet was detected.
- RadiusUnsupportedAttribute - The request contained an unsupported attribute.
- RadiusMissingAttribute - A request was missing a required attribute.
- RadiusNasIdMismatch - The system was unable to match the received NAS to the stored information.
- RadiusInvalidRequest - The system has received an invalid request.
- RadiusUnsupportedService - The requested service is not supported.
- RadiusUnsupportedExtension - The requested extension is not supported.
- RadiusInvalidAttributeValue - The request contains an invalid attribute value.
- RadiusAdministrativelyProhibited - The request is administratively prohibited.
- RadiusRequestNotRoutable - The request can't be routed.
- RadiusSessionNotFound - The session context cannot be found.
- RadiusSessionNotRemoveable - The session context cannot be removed.
- RadiusProxyProcessingError - An unknown proxy processing error has occurred.
- RadiusResourcesUnavailable - The necessary resources are unavailable.
- RadiusRequestInitiated - A request has been initiated.
- RadiusMultiSessionSelectionUnsupported - The requested multiple-session selection is not supported.
- RadiusLocationInfoRequired - The location information is missing.

```
<?xml version='1.0' ?>
<Statistics>
 <ProtocolErrorStats>
   <Sample>
     <StartTime>2011-02-22T03:45:11Z</StartTime>
     <EndTime>2011-02-22T03:45:33Z</EndTime>
     <PolicyServer>rc178</PolicyServer>
     <DiameterCommandUnsupportedReceived>0</DiameterCommandUnsupportedReceived>
     <DiameterCommandUnsupportedSent>0</DiameterCommandUnsupportedSent>
     <DiameterUnableToDeliverReceived>0</DiameterUnableToDeliverReceived>
     <DiameterUnableToDeliverSent>0</DiameterUnableToDeliverSent>
```

```
    <DiameterRealmNotServedReceived>0</DiameterRealmNotServedReceived>
    <DiameterRealmNotServedSent>0</DiameterRealmNotServedSent>
    <DiameterTooBusyReceived>0</DiameterTooBusyReceived>
    <DiameterTooBusySent>0</DiameterTooBusySent>
    <DiameterLoopDetectedReceived>0</DiameterLoopDetectedReceived>
    <DiameterLoopDetectedSent>0</DiameterLoopDetectedSent>
    <DiameterRedirectIndicationReceived>0</DiameterRedirectIndicationReceived>
    <DiameterRedirectIndicationSent>0</DiameterRedirectIndicationSent>
    <DiameterApplicationUnsupportedReceived>0</DiameterApplication
       UnsupportedReceived>
    <DiameterApplicationUnsupportedSent>0</DiameterApplicationUnsupportedSent>
    <DiameterInvalidHdrBitsReceived>0</DiameterInvalidHdrBitsReceived>
    <DiameterInvalidHdrBitsSent>0</DiameterInvalidHdrBitsSent>
    <DiameterInvalidAvpBitsReceived>0</DiameterInvalidAvpBitsReceived>
    <DiameterInvalidAvpBitsSent>0</DiameterInvalidAvpBitsSent>
    <DiameterUnknownPeerReceived>0</DiameterUnknownPeerReceived>
    <DiameterUnknownPeerSent>0</DiameterUnknownPeerSent>
    <DiameterAuthenticationRejectedReceived>0</DiameterAuthentication
       RejectedReceived>
    <DiameterAuthenticationRejectedSent>0</DiameterAuthentication
       RejectedSent>
    <DiameterOutOfSpaceReceived>0</DiameterOutOfSpaceReceived>
    <DiameterOutOfSpaceSent>0</DiameterOutOfSpaceSent>
    <ElectionLostReceived>0</ElectionLostReceived>
    <ElectionLostSent>0</ElectionLostSent>
    <DiameterEndUserServiceDeniedReceived>0</DiameterEndUserService
       DeniedReceived>
    <DiameterEndUserServiceDeniedSent>0</DiameterEndUserServiceDeniedSent>
    <DiameterCreditControlNotApplicableReceived>0</DiameterCreditControlNot
       ApplicableReceived>
    <DiameterCreditControlNotApplicableSent>0</DiameterCreditControlNot
       ApplicableSent>
    <DiameterCreditLimitReachedReceived>0</DiameterCreditLimitReachedReceived>
    <DiameterCreditLimitReachedSent>0</DiameterCreditLimitReachedSent>
    <DiameterAvpUnsupportedReceived>0</DiameterAvpUnsupportedReceived>
    <DiameterAvpUnsupportedSent>0</DiameterAvpUnsupportedSent>
    <DiameterUnknownSessionIdReceived>0</DiameterUnknownSessionIdReceived>
    <DiameterUnknownSessionIdSent>0</DiameterUnknownSessionIdSent>
    <DiameterAuthorizationRejectedReceived>0</DiameterAuthorization
       RejectedReceived>
    <DiameterAuthorizationRejectedSent>0</DiameterAuthorizationRejectedSent>
    <DiameterInvalidAvpValueReceived>0</DiameterInvalidAvpValueReceived>
    <DiameterInvalidAvpValueSent>0</DiameterInvalidAvpValueSent>
    <DiameterMissingAvpReceived>0</DiameterMissingAvpReceived>
    <DiameterMissingAvpSent>0</DiameterMissingAvpSent>
    <DiameterResourcesExceededReceived>0</DiameterResourcesExceededReceived>
    <DiameterResourcesExceededSent>0</DiameterResourcesExceededSent>
    <DiameterContradictingAvpsReceived>0</DiameterContradictingAvpsReceived>
    <DiameterContradictingAvpsSent>0</DiameterContradictingAvpsSent>
    <DiameterAvpNotAllowedReceived>0</DiameterAvpNotAllowedReceived>
    <DiameterAvpNotAllowedSent>0</DiameterAvpNotAllowedSent>
    <DiameterAvpOccursTooManyTimesReceived>0</DiameterAvpOccursTooMany
       TimesReceived>
    <DiameterAvpOccursTooManyTimesSent>0</DiameterAvpOccursTooManyTimesSent>
    <DiameterNoCommonApplicationReceived>0</DiameterNoCommonApplicationReceived>
    <DiameterNoCommonApplicationSent>0</DiameterNoCommonApplicationSent>
    <DiameterUnsupportedVersionReceived>0</DiameterUnsupportedVersionReceived>
    <DiameterUnsupportedVersionSent>0</DiameterUnsupportedVersionSent>
    <DiameterUnableToComplyReceived>0</DiameterUnableToComplyReceived>
    <DiameterUnableToComplySent>0</DiameterUnableToComplySent>
    <DiameterInvalidBitInHeaderReceived>0</DiameterInvalidBitInHeaderReceived>
    <DiameterInvalidBitInHeaderSent>0</DiameterInvalidBitInHeaderSent>
    <DiameterInvalidAvpLengthReceived>0</DiameterInvalidAvpLengthReceived>
    <DiameterInvalidAvpLengthSent>0</DiameterInvalidAvpLengthSent>
   <DiameterInvalidMessageLengthReceived>0</DiameterInvalidMessageLengthReceived>
```

```
    <DiameterInvalidMessageLengthSent>0</DiameterInvalidMessageLengthSent>
    <DiameterInvalidAvpBitComboReceived>0</DiameterInvalidAvpBitComboReceived>
    <DiameterInvalidAvpBitComboSent>0</DiameterInvalidAvpBitComboSent>
    <DiameterNoCommonSecurityReceived>0</DiameterNoCommonSecurityReceived>
    <DiameterNoCommonSecuritySent>0</DiameterNoCommonSecuritySent>
    <DiameterUserUnknownReceived>0</DiameterUserUnknownReceived>
    <DiameterUserUnknownSent>0</DiameterUserUnknownSent>
    <DiameterRatingFailedReceived>0</DiameterRatingFailedReceived>
    <DiameterRatingFailedSent>0</DiameterRatingFailedSent>
    <DiameterErrorInitialParametersReceived>0</DiameterErrorInitial
       ParametersReceived>
    <DiameterErrorInitialParametersSent>2</DiameterErrorInitialParametersSent>
    <RadiusSessionContyextRemovedReceived>0</RadiusSessionContyext
       RemovedReceived>
    <RadiusSessionContyextRemovedSent>0</RadiusSessionContyextRemovedSent>
    <RadiusInvalidEapPacketReceived>0</RadiusInvalidEapPacketReceived>
    <RadiusInvalidEapPacketSent>0</RadiusInvalidEapPacketSent>
    <RadiusUnsupportedAttributeReceived>0</RadiusUnsupportedAttributeReceived>
    <RadiusUnsupportedAttributeSent>0</RadiusUnsupportedAttributeSent>
    <RadiusMissingAttributeReceived>0</RadiusMissingAttributeReceived>
    <RadiusMissingAttributeSent>0</RadiusMissingAttributeSent>
    <RadiusNasIdMismatchReceived>0</RadiusNasIdMismatchReceived>
    <RadiusNasIdMismatchSent>0</RadiusNasIdMismatchSent>
    <RadiusInvalidRequestReceived>0</RadiusInvalidRequestReceived>
    <RadiusInvalidRequestSent>0</RadiusInvalidRequestSent>
    <RadiusUnsupportedServiceReceived>0</RadiusUnsupportedServiceReceived>
    <RadiusUnsupportedServiceSent>0</RadiusUnsupportedServiceSent>
    <RadiusUnsupportedExtensionReceived>0</RadiusUnsupportedExtensionReceived>
    <RadiusUnsupportedExtensionSent>0</RadiusUnsupportedExtensionSent>
    <RadiusInvalidAttributeValueReceived>0</RadiusInvalidAttributeValueReceived>
    <RadiusInvalidAttributeValueSent>0</RadiusInvalidAttributeValueSent>
    <RadiusAdministrativelyProhibitedReceived>0</RadiusAdministratively
       ProhibitedReceived>
    <RadiusAdministrativelyProhibitedSent>0</RadiusAdministratively
       ProhibitedSent>
    <RadiusRequestNotRoutableReceived>0</RadiusRequestNotRoutableReceived>
    <RadiusRequestNotRoutableSent>0</RadiusRequestNotRoutableSent>
    <RadiusSessionNotFoundReceived>0</RadiusSessionNotFoundReceived>
    <RadiusSessionNotFoundSent>0</RadiusSessionNotFoundSent>
    <RadiusSessionNotRemoveableReceived>0</RadiusSessionNotRemoveableReceived>
    <RadiusSessionNotRemoveableSent>0</RadiusSessionNotRemoveableSent>
    <RadiusProxyProcessingErrorReceived>0</RadiusProxyProcessingErrorReceived>
    <RadiusProxyProcessingErrorSent>0</RadiusProxyProcessingErrorSent>
    <RadiusResourcesUnavailableReceived>0</RadiusResourcesUnavailableReceived>
    <RadiusResourcesUnavailableSent>0</RadiusResourcesUnavailableSent>
    <RadiusRequestInitiatedReceived>0</RadiusRequestInitiatedReceived>
    <RadiusRequestInitiatedSent>0</RadiusRequestInitiatedSent>
    <RadiusMultiSessionSelectionUnsupportedReceived>0</RadiusMultiSession
       SelectionUnsupportedReceived>
    <RadiusMultiSessionSelectionUnsupportedSent>0</RadiusMultiSessionSelection
       UnsupportedSent>
    <RadiusLocationInfoRequiredReceived>0</RadiusLocationInfoRequiredReceived>
    <RadiusLocationInfoRequiredSent>0</RadiusLocationInfoRequiredSent>
  </Sample>
```

```
  </ProtocolErrorStats>
</Statistics>
```

Session Cleanup Statistics

The following examples show the request and response that are defined in the XSDs for the StaleSessionStats tag.

Request

This request follows the **QueryOmStats ➤ StaleSessionStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2001-12-31T12:00:00</StartTime>
    <EndTime>2001-12-31T12:15:00</EndTime>
    <StaleSessionStats>
      <PolicyServer>PolicyServer</PolicyServer>
      </ProtocolErrorStats>
    </StaleSessionStats>
</QueryOmStats>
```

The response to this request follows the **Statistics ➤ StaleSessionStats** Stag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <staleSessionStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <SessionReadyForCleanUp>0</SessionReadyForCleanUp>
      <SessionCleaned>0</SessionCleaned>
      <ReauthorizedSessions>0</ReauthorizedSessions>
      <ReauthorizedTimeouts>0</ReauthorizedTimeouts>
      <ReauthorizedCleaned>0</ReauthorizedCleaned>
  </StaleSessionStats>
```

Individual statistics are defined as follows:

- **Ready for Cleanup**: GAUGE to track the number of current stale sessions.
- **Removed for Error**: COUNTER to track the number of deleted stale sessions that failed to reauthorize due to an error response.
- **Reauthorized**: COUNTER to track the number of reauthorized stale sessions.
- **Reauthorization Timeouts**: COUNTER to track to the number of timed out reauthorization sessions.
- **Removed for Expiration**: COUNTER to track the number of timed out reauthorization sessions due to age.

Timeout Statistics

The following examples show the response defined in the XSDs for the DiameterAf Stats tag.

The MRA response follows the **Statistics ➤ DiameterAfStats** tag defined in the XSDs.

- **AAR Timeout**: The numeric count of AA-Request messages sent and AA-Answer messages not received in a set amount of time.
- **ASR Timeout**: The numeric count of Abort Session Requests (ASR) messages sent where ASA messages not received in a set amount of time.
- **RAR Timeout**:The numeric count of ReAuthorization Request (RAR) messages sent where RAA messages not received in a set amount of time.
- **STR Timeout**:The numeric count of Session Termination Requests (STR) messages sent where STA messages not received in a set amount of time.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterAfStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      ...
      <ASRMessagesReceivedCount>0</ASRMessagesReceivedCount>
      <ASRMessagesSentCount>0</ASRMessagesSentCount>
      <ASRMessagesTimeoutCount>0</ASRMessagesTimeoutCount>
      ...
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
      ...
      <STRMessagesReceivedCount>0</STRMessagesReceivedCount>
      <STRMessagesSentCount>0</STRMessagesSentCount>
      <STRMessagesTimeoutCount>0</STRMessagesTimeoutCount>
      ...
      <AARMessagesReceivedCount>0</AARMessagesReceivedCount>
      <AARMessagesSentCount>0</AARMessagesSentCount>
      <AARMessagesTimeoutCount>0</AARMessagesTimeoutCount>
      ...
      <PeerReopenCount>0</PeerReopenCount>
    </Sample>
  </DiameterAfStats>
</Statistics>
```

The MRA response follows the **Statistics ➤ DiameterAfPeerStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterAfPeerStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <Name>NetworkElementName></Name>
```

```
      <NeId>NeId</NeId>
      <NetworkElementType>NetworkElementType</NetworkElementType>
      <NetworkElementSubType>NetworkElementSubType</NetworkElementSubType>
      ...
      <ASRMessagesReceivedCount>0</ASRMessagesReceivedCount>
      <ASRMessagesSentCount>0</ASRMessagesSentCount>
      <ASRMessagesTimeoutCount>0</ASRMessagesTimeoutCount>
      ...
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
      ...
     <STRMessagesReceivedCount>STRMessagesReceivedCount</STRMessagesReceivedCount>

      <STRMessagesSentCount>STRMessagesSentCount</STRMessagesSentCount>
      <STRMessagesTimeoutCount>0</STRMessagesTimeoutCount>
      ...
     <AARMessagesReceivedCount>AARMessagesReceivedCount</AARMessagesReceivedCount>

      <AARMessagesSentCount>AARMessagesSentCount</AARMessagesSentCount>
      <AARMessagesTimeoutCount>0</AARMessagesTimeoutCount>
      ...
      <AAAFailureMessagesSentCount>0</AAAFailureMessagesSentCount>
    </Sample>
  </DiameterAfPeerStats>
</Statistics>
```

The MRA response follows the **Statistics ➤ DiameterPcefStats** tag defined in the XSDs.

- **CCR Timeout**: The numeric count of CC-Request messages sent and CC-Answer messages not received in a set amount of time.

  Note: The number of CCR-I/U/T Timeouts should equal this number combined.

- **CCR-I Timeout**: The numeric count of CCR-Initial messages sent and CCA-Initial messages not received in a set amount of time.
- **CCR-U Timeout**: The numeric count of CCR-Update messages sent and CCA-Update messages not received in a set amount of time.
- **CCR-T Timeout**: The numeric count of CCR-Terminate messages sent and CCA-Terminate messages not received in a set amount of time.
- **RAR Timeout**: The numeric count of Reauthorization Request messages sent and RAA messages not received in a set amount of time.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterPcefStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <CurrentConnectionsCount>0</CurrentConnectionsCount>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
      ...
      <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
      <CCRMessagesSentCount>0</CCRMessagesSentCount>
      <CCRMessagesTimeoutCount>0</CCRMessagesTimeoutCount>
```

```
      <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
      <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
      <CCRIMessagesTimeoutCount>0</CCRIMessagesTimeoutCount>
      <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
      <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
      <CCRUMessagesTimeoutCount>0</CCRUMessagesTimeoutCount>
      <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
      <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
      <CCRTMessagesTimeoutCount>0</CCRTMessagesTimeoutCount>
      ...
      <PeerSuspectCount>0</PeerSuspectCount>
      <PeerReopenCount>0</PeerReopenCount>
    </Sample>
  </DiameterPcefStats>
</Statistics>
```

The MRA response follows the **Statistics ➤ DiameterPcefPeerStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterPcefPeerStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <Name>NetworkElementName</Name>
      <NeId>NeId<NeId>
      <NetworkElementType>NetworkElementType</NetworkElementType>
      <NetworkElementSubType>NetworkElementSubType</NetworkElementSubType>
      ...
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
      ...
      <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
      <CCRMessagesSentCount>0</CCRMessagesSentCount>
      <CCRMessagesTimeoutCount>0</CCRMessagesTimeoutCount>
      <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
      <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
      <CCRIMessagesTimeoutCount>0</CCRIMessagesTimeoutCount>
      <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
      <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
      <CCRUMessagesTimeoutCount>0</CCRUMessagesTimeoutCount>
      <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
      <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
      <CCRTMessagesTimeoutCount>0</CCRTMessagesTimeoutCount>
      ...
      <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
    </Sample>
  </DiameterPcefPeerStats>
</Statistics>
```

The MRA response follows the **Statistics ➤ DiameterBberfStats** tag defined in the XSDs.

- **CCR Timeout**: The numeric count of CC-Request messages sent and CC-Answer messages not received in a set amount of time.

  Note: The number of CCR-I/U/T Timeouts should equal this number combined.

- **CCR-I Timeout**: The numeric count of CCR-Initial messages sent and CCA-Initial messages not received in a set amount of time.

- **CCR-U Timeout**: The numeric count of CCR-Update messages sent and CCA-Update messages not received in a set amount of time.
- **CCR-T Timeout**: The numeric count of CCR-Terminate messages sent and CCA-Terminate messages not received in a set amount of time.
- **RAR Timeout**: The numeric count of Reauthorization Request messages sent and RAA messages not received in a set amount of time.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterBberfStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <CurrentConnectionsCount>0</CurrentConnectionsCount>
      ...
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
      ...
      <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
      <CCRMessagesSentCount>0</CCRMessagesSentCount>
      <CCRMessagesTimeoutCount>0</CCRMessagesTimeoutCount>
      <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
      <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
      <CCRIMessagesTimeoutCount>0</CCRIMessagesTimeoutCount>
      <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
      <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
      <CCRUMessagesTimeoutCount>0</CCRUMessagesTimeoutCount>
      <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
      <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
      <CCRTMessagesTimeoutCount>0</CCRTMessagesTimeoutCount>
      ...
      <PeerReopenCount>0</PeerReopenCount>
    </Sample>
  </DiameterBberfStats>
</Statistics>
```

The following is an example of a response for an MPE:

```
<Statistics>
  <DiameterBberfStats>
    <Sample>
    <StartTime>2012-06-07T01:00:00Z</StartTime>
    <EndTime>2012-06-07T01:15:00Z</EndTime>
    <PolicyServer>kli-63-mpe</PolicyServer>
    <IsComplete>true</IsComplete>
    <CurrentConnectionsCount>1</CurrentConnectionsCount>
    <MessagesInCount>0</MessagesInCount>
    <MessagesOutCount>0</MessagesOutCount>
    <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
    <RARMessagesSentCount>0</RARMessagesSentCount>
    <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
    <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
    <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
    <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
    <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
    <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
    <CCRMessagesSentCount>0</CCRMessagesSentCount>
    <CCRMessagesTimeoutCount>0</CCRMessagesTimeoutCount>
```

```
    <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
    <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
    <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
    <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
    <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
    <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
    <CCRIMessagesTimeoutCount>0</CCRIMessagesTimeoutCount>
    <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
    <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
    <CCRUMessagesTimeoutCount>0</CCRUMessagesTimeoutCount>
    <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
    <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
    <CCRTMessagesTimeoutCount>0</CCRTMessagesTimeoutCount>
    <CCAISuccessMessagesReceivedCount>0</CCAISuccessMessagesReceivedCount>
    <CCAISuccessMessagesSentCount>0</CCAISuccessMessagesSentCount>
    <CCAIFailureMessagesReceivedCount>0</CCAIFailureMessagesReceivedCount>
    <CCAIFailureMessagesSentCount>0</CCAIFailureMessagesSentCount>
    <CCAUSuccessMessagesReceivedCount>0</CCAUSuccessMessagesReceivedCount>
    <CCAUSuccessMessagesSentCount>0</CCAUSuccessMessagesSentCount>
    <CCAUFailureMessagesReceivedCount>0</CCAUFailureMessagesReceivedCount>
    <CCAUFailureMessagesSentCount>0</CCAUFailureMessagesSentCount>
    <CCATSuccessMessagesReceivedCount>0</CCATSuccessMessagesReceivedCount>
    <CCATSuccessMessagesSentCount>0</CCATSuccessMessagesSentCount>
    <CCATFailureMessagesReceivedCount>0</CCATFailureMessagesReceivedCount>
    <CCATFailureMessagesSentCount>0</CCATFailureMessagesSentCount>
    <ActiveSessionsCount>0</ActiveSessionsCount>
    <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
    <PeerOkayCount>1</PeerOkayCount>
    <PeerDownCount>0</PeerDownCount>
    <PeerSuspectCount>0</PeerSuspectCount>
    <PeerReopenCount>0</PeerReopenCount>
    </Sample>
  </DiameterBberfStats>
</Statistics>
```

The MRA response follows the **Statistics ➤ DiameterBberfPeerStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterBberfPeerStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <Name>NetworkElementName</Name>
      <NeId>NeId</NeId>
      <NetworkElementType>NetworkElementType</NetworkElementType>
      <NetworkElementSubType>NetworkElementSubType</NetworkElementSubType>
      ...
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
      ...
      <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
      <CCRMessagesSentCount>0</CCRMessagesSentCount>
      <CCRMessagesTimeoutCount>0</CCRMessagesTimeoutCount>
      <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
      <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
      <CCRIMessagesTimeoutCount>0</CCRIMessagesTimeoutCount>
      <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
      <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
      <CCRUMessagesTimeoutCount>0</CCRUMessagesTimeoutCount>
```

```
      <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
      <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
      <CCRTMessagesTimeoutCount>0</CCRTMessagesTimeoutCount>
      ...
      <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
    </Sample>
  </DiameterBberfPeerStats>
</Statistics>
```

The following is an example of a response for an MPE:

```
<Statistics>
  <DiameterBberfPeerStats>
    <Sample>
    <StartTime>2012-06-07T01:00:00Z</StartTime>
    <EndTime>2012-06-07T01:15:00Z</EndTime>
    <PolicyServer>kli-63-mpe</PolicyServer>
    <IsComplete>true</IsComplete>
    <Name>mra66</Name>
    <NeId />
    <NetworkElementType />
    <NetworkElementSubType />
    <ConnectTime>Tue Jun 05 02:54:29 EDT 2012</ConnectTime>
    <DisconnectTime>Tue Jun 05 02:54:28 EDT 2012</DisconnectTime>
    <ConnectAddress>10.60.25.66</ConnectAddress>
    <ConnectPort>48526</ConnectPort>
    <ConnectType>TCP</ConnectType>
    <MessagesInCount>0</MessagesInCount>
    <MessagesOutCount>0</MessagesOutCount>
    <MessagesErrorInCount>0</MessagesErrorInCount>
    <MessagesErrorOutCount>0</MessagesErrorOutCount>
    <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
    <RARMessagesSentCount>0</RARMessagesSentCount>
    <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
    <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
    <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
    <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
    <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
    <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
    <CCRMessagesSentCount>0</CCRMessagesSentCount>
    <CCRMessagesTimeoutCount>0</CCRMessagesTimeoutCount>
    <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
    <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
    <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
    <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
    <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
    <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
    <CCRIMessagesTimeoutCount>0</CCRIMessagesTimeoutCount>
    <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
    <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
    <CCRUMessagesTimeoutCount>0</CCRUMessagesTimeoutCount>
    <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
    <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
    <CCRTMessagesTimeoutCount>0</CCRTMessagesTimeoutCount>
    <CCAISuccessMessagesReceivedCount>0</CCAISuccessMessagesReceivedCount>
    <CCAISuccessMessagesSentCount>0</CCAISuccessMessagesSentCount>
    <CCAIFailureMessagesReceivedCount>0</CCAIFailureMessagesReceivedCount>
    <CCAIFailureMessagesSentCount>0</CCAIFailureMessagesSentCount>
    <CCAUSuccessMessagesReceivedCount>0</CCAUSuccessMessagesReceivedCount>
    <CCAUSuccessMessagesSentCount>0</CCAUSuccessMessagesSentCount>
    <CCAUFailureMessagesReceivedCount>0</CCAUFailureMessagesReceivedCount>
    <CCAUFailureMessagesSentCount>0</CCAUFailureMessagesSentCount>
    <CCATSuccessMessagesReceivedCount>0</CCATSuccessMessagesReceivedCount>
    <CCATSuccessMessagesSentCount>0</CCATSuccessMessagesSentCount>
```

```
    <CCATFailureMessagesReceivedCount>0</CCATFailureMessagesReceivedCount>
    <CCATFailureMessagesSentCount>0</CCATFailureMessagesSentCount>
    <ActiveSessionsCount>0</ActiveSessionsCount>
    <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
   </Sample>
 </DiameterBberfPeerStats>
</Statistics>
```

The MRA response follows the **Statistics ➤ DiameterShStats** tag defined in the XSDs.

- **UDR Timeout**: The numeric count of User Data Request (UDR) messages sent and User Data Answer (UDA) messages not received in a set amount of time.
- **PUR Timeout**:The numeric count of Profile Update Request (PUR) messages sent and Profile Update Answer (PUA) messages not received in a set amount of time.
- **SNR Timeout**:The numeric count of Subscription Notification Request (SNR) messages sent and Subscription Notification Answer (SNA) messages not received in a set amount of time.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <DiameterShStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <CurrentConnectionsCount>0</CurrentConnectionsCount>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <UDRMessagesReceivedCount>0</UDRMessagesReceivedCount>
      <UDRMessagesSentCount>0</UDRMessagesSentCount>
      <UDRMessagesTimeoutCount>0</UDRMessagesTimeoutCount>
      ...
      <PURMessagesReceivedCount>0</PURMessagesReceivedCount>
      <PURMessagesSentCount>0</PURMessagesSentCount>
      <PURMessagesTimeoutCount>0</PURMessagesTimeoutCount>
      ...
      <SNRMessagesReceivedCount>0</SNRMessagesReceivedCount>
      <SNRMessagesSentCount>0</SNRMessagesSentCount>
      <SNRMessagesTimeoutCount>0</SNRMessagesTimeoutCount>
      ...
      <PeerReopenCount>0</PeerReopenCount>
   </Sample>
  </DiameterShStats>
</Statistics>
```

The MRA response follows the **Statistics ➤ DiameterTdfStats** tag defined in the XSDs.

CCR Timeout: The numeric count of CC-Request messages sent and CC-Answer messages not received in a set amount of time.

Note: The number of CCR-I/U/T Timeouts should equal this number combined.

- **CCR-I Timeout**: The numeric count of CCR-Initial messages sent and CCA-Initial messages not received in a set amount of time.
- **CCR-U Timeout**: The numeric count of CCR-Update messages sent and CCA-Update messages not received in a set amount of time.
- **CCR-T Timeout**:The numeric count of CCR-Terminate messages sent and CCA-Terminate messages not received in a set amount of time.

- **RAR Timeout**: The numeric count of Reauthorization Request messages sent and RAA messages not received in a set amount of time.

The following is an example of a response for a single policy server:

```
<Statistics>
  <DiameterTdfStats>
    <Sample>
      <StartTime>2012-05-28T09:30:00Z</StartTime>
      <EndTime>2012-05-28T09:45:00Z</EndTime>
      <PolicyServer>kli-63-mpe</PolicyServer>
      <IsComplete>true</IsComplete>
      <CurrentConnectionsCount>1</CurrentConnectionsCount>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <TSRMessagesReceivedCount>0</TSRMessagesReceivedCount>
      <TSRMessagesSentCount>0</TSRMessagesSentCount>
      <TSRMessagesTimeoutCount>0</TSRMessagesTimeoutCount>
      <TSASuccessMessagesReceivedCount>0</TSASuccessMessagesReceivedCount>
      <TSASuccessMessagesSentCount>0</TSASuccessMessagesSentCount>
      <TSAFailureMessagesReceivedCount>0</TSAFailureMessagesReceivedCount>
      <TSAFailureMessagesSentCount>0</TSAFailureMessagesSentCount>
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
      <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
      <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
      <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
      <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
      <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
      <CCRMessagesSentCount>0</CCRMessagesSentCount>
      <CCRMessagesTimeoutCount>0</CCRMessagesTimeoutCount>
      <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
      <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
      <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
      <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
      <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
      <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
      <CCRIMessagesTimeoutCount>0</CCRIMessagesTimeoutCount>
      <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
      <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
      <CCRUMessagesTimeoutCount>0</CCRUMessagesTimeoutCount>
      <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
      <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
      <CCRTMessagesTimeoutCount>0</CCRTMessagesTimeoutCount>
      <CCAISuccessMessagesReceivedCount>0</CCAISuccessMessagesReceivedCount>
      <CCAISuccessMessagesSentCount>0</CCAISuccessMessagesSentCount>
      <CCAIFailureMessagesReceivedCount>0</CCAIFailureMessagesReceivedCount>
      <CCAIFailureMessagesSentCount>0</CCAIFailureMessagesSentCount>
      <CCAUSuccessMessagesReceivedCount>0</CCAUSuccessMessagesReceivedCount>
      <CCAUSuccessMessagesSentCount>0</CCAUSuccessMessagesSentCount>
      <CCAUFailureMessagesReceivedCount>0</CCAUFailureMessagesReceivedCount>
      <CCAUFailureMessagesSentCount>0</CCAUFailureMessagesSentCount>
      <CCATSuccessMessagesReceivedCount>0</CCATSuccessMessagesReceivedCount>
      <CCATSuccessMessagesSentCount>0</CCATSuccessMessagesSentCount>
      <CCATFailureMessagesReceivedCount>0</CCATFailureMessagesReceivedCount>
      <CCATFailureMessagesSentCount>0</CCATFailureMessagesSentCount>
      <ActiveSessionsCount>0</ActiveSessionsCount>
      <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
      <PeerOkayCount>1</PeerOkayCount>
      <PeerDownCount>2</PeerDownCount>
      <PeerSuspectCount>0</PeerSuspectCount>
      <PeerReopenCount>0</PeerReopenCount>
    </Sample>
```

```
  </DiameterTdfStats>
</Statistics>
```

The following is an example of a response for an MPE:

```
<Statistics>
  <DiameterTdfStats>
     <Sample>
     <StartTime>2012-05-28T09:30:00Z</StartTime>
     <EndTime>2012-05-28T09:45:00Z</EndTime>
     <PolicyServer>kli-63-mpe</PolicyServer>
     <IsComplete>true</IsComplete>
     <CurrentConnectionsCount>1</CurrentConnectionsCount>
     <MessagesInCount>0</MessagesInCount>
     <MessagesOutCount>0</MessagesOutCount>
     <TSRMessagesReceivedCount>0</TSRMessagesReceivedCount>
     <TSRMessagesSentCount>0</TSRMessagesSentCount>
     <TSRMessagesTimeoutCount>0</TSRMessagesTimeoutCount>
     <TSASuccessMessagesReceivedCount>0</TSASuccessMessagesReceivedCount>
     <TSASuccessMessagesSentCount>0</TSASuccessMessagesSentCount>
     <TSAFailureMessagesReceivedCount>0</TSAFailureMessagesReceivedCount>
     <TSAFailureMessagesSentCount>0</TSAFailureMessagesSentCount>
     <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
     <RARMessagesSentCount>0</RARMessagesSentCount>
     <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
     <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
     <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
     <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
     <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
     <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
     <CCRMessagesSentCount>0</CCRMessagesSentCount>
     <CCRMessagesTimeoutCount>0</CCRMessagesTimeoutCount>
     <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
     <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
     <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
     <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
     <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
     <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
     <CCRIMessagesTimeoutCount>0</CCRIMessagesTimeoutCount>
     <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
     <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
     <CCRUMessagesTimeoutCount>0</CCRUMessagesTimeoutCount>
     <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
     <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
     <CCRTMessagesTimeoutCount>0</CCRTMessagesTimeoutCount>
     <CCAISuccessMessagesReceivedCount>0</CCAISuccessMessagesReceivedCount>
     <CCAISuccessMessagesSentCount>0</CCAISuccessMessagesSentCount>
     <CCAIFailureMessagesReceivedCount>0</CCAIFailureMessagesReceivedCount>
     <CCAIFailureMessagesSentCount>0</CCAIFailureMessagesSentCount>
     <CCAUSuccessMessagesReceivedCount>0</CCAUSuccessMessagesReceivedCount>
     <CCAUSuccessMessagesSentCount>0</CCAUSuccessMessagesSentCount>
     <CCAUFailureMessagesReceivedCount>0</CCAUFailureMessagesReceivedCount>
     <CCAUFailureMessagesSentCount>0</CCAUFailureMessagesSentCount>
     <CCATSuccessMessagesReceivedCount>0</CCATSuccessMessagesReceivedCount>
     <CCATSuccessMessagesSentCount>0</CCATSuccessMessagesSentCount>
     <CCATFailureMessagesReceivedCount>0</CCATFailureMessagesReceivedCount>
     <CCATFailureMessagesSentCount>0</CCATFailureMessagesSentCount>
     <ActiveSessionsCount>0</ActiveSessionsCount>
     <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
     <PeerOkayCount>1</PeerOkayCount>
     <PeerDownCount>2</PeerDownCount>
     <PeerSuspectCount>0</PeerSuspectCount>
     <PeerReopenCount>0</PeerReopenCount>
     </Sample>
```

```
   </DiameterTdfStats>
</Statistics>
```

The MRA response follows the **Statistics ➤ DiameterTdfPeerStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<Statistics>
  <DiameterTdfPeerStats>
    <Sample>
      <StartTime>2012-05-28T09:15:00Z</StartTime>
      <EndTime>2012-05-28T09:30:00Z</EndTime>
      <PolicyServer>kli-63-mpe</PolicyServer>
      <IsComplete>true</IsComplete>
      <Name>kli-66-mra2</Name>
      <NeId/>
      <NetworkElementType/>
      <NetworkElementSubType/>
      <ConnectTime>Thu May 24 01:23:51 EDT 2012</ConnectTime>
      <DisconnectTime>Fri May 25 04:18:11 EDT 2012</DisconnectTime>
      <ConnectAddress>10.60.25.67:57455,10.60.25.68:57455</ConnectAddress>
      <ConnectPort>0</ConnectPort>
      <ConnectType>SCTP</ ConnectType>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <MessagesErrorInCount>0</MessagesErrorInCount>
      <MessagesErrorOutCount>0</MessagesErrorOutCount>
      <TSRMessagesReceivedCount>0</TSRMessagesReceivedCount>
      <TSRMessagesSentCount>0</TSRMessagesSentCount>
      <TSRMessagesTimeoutCount>0</TSRMessagesTimeoutCount>
      <TSASuccessMessagesReceivedCount>0</TSASuccessMessagesReceivedCount>
      <TSASuccessMessagesSentCount>0</TSASuccessMessagesSentCount>
      <TSAFailureMessagesReceivedCount>0</TSAFailureMessagesReceivedCount>
      <TSAFailureMessagesSentCount>0</TSAFailureMessagesSentCount>
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
      <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
      <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
      <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
      <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
      <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
      <CCRMessagesSentCount>0</CCRMessagesSentCount>
      <CCRMessagesTimeoutCount>0</CCRMessagesTimeoutCount>
      <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
      <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
      <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
      <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
      <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
      <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
      <CCRIMessagesTimeoutCount>0</CCRIMessagesTimeoutCount>
      <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
      <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
      <CCRUMessagesTimeoutCount>0</CCRUMessagesTimeoutCount>
      <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
      <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
      <CCRTMessagesTimeoutCount>0</CCRTMessagesTimeoutCount>
      <CCAISuccessMessagesReceivedCount>0</CCAISuccessMessagesReceivedCount>
      <CCAISuccessMessagesSentCount>0</CCAISuccessMessagesSentCount>
      <CCAIFailureMessagesReceivedCount>0</CCAIFailureMessagesReceivedCount>
      <CCAIFailureMessagesSentCount>0</CCAIFailureMessagesSentCount>
      <CCAUSuccessMessagesReceivedCount>0</CCAUSuccessMessagesReceivedCount>
      <CCAUSuccessMessagesSentCount>0</CCAUSuccessMessagesSentCount>
      <CCAUFailureMessagesReceivedCount>0</CCAUFailureMessagesReceivedCount>
```

```
      <CCAUFailureMessagesSentCount>0</CCAUFailureMessagesSentCount>
      <CCATSuccessMessagesReceivedCount>0</CCATSuccessMessagesReceivedCount>
      <CCATSuccessMessagesSentCount>0</CCATSuccessMessagesSentCount>
      <CCATFailureMessagesReceivedCount>0</CCATFailureMessagesReceivedCount>
      <CCATFailureMessagesSentCount>0</CCATFailureMessagesSentCount>
      <ActiveSessionsCount>0</ActiveSessionsCount>
      <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
    </Sample>
  </DiameterTdfPeerStats>
</Statistics>
```

The following is an example of a response for an MPE:

```
<Statistics>
   <DiameterTdfPeerStats>
      <Sample>
      <StartTime>2012-05-28T09:15:00Z</StartTime>
      <EndTime>2012-05-28T09:30:00Z</EndTime>
      <PolicyServer>kli-63-mpe</PolicyServer>
      <IsComplete>true</IsComplete>
      <Name>kli-66-mra2</Name>
      <NeId />
      <NetworkElementType />
      <NetworkElementSubType />
      <ConnectTime>Thu May 24 01:23:51 EDT 2012</ConnectTime>
      <DisconnectTime>Fri May 25 04:18:11 EDT 2012</DisconnectTime>
      <ConnectAddress>10.60.25.67:57455,10.60.25.68:57455</ConnectAddress>
      <ConnectPort>0</ConnectPort>
      <ConnectType />
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <MessagesErrorInCount>0</MessagesErrorInCount>
      <MessagesErrorOutCount>0</MessagesErrorOutCount>
      <TSRMessagesReceivedCount>0</TSRMessagesReceivedCount>
      <TSRMessagesSentCount>0</TSRMessagesSentCount>
      <TSRMessagesTimeoutCount>0</TSRMessagesTimeoutCount>
      <TSASuccessMessagesReceivedCount>0</TSASuccessMessagesReceivedCount>
      <TSASuccessMessagesSentCount>0</TSASuccessMessagesSentCount>
      <TSAFailureMessagesReceivedCount>0</TSAFailureMessagesReceivedCount>
      <TSAFailureMessagesSentCount>0</TSAFailureMessagesSentCount>
      <RARMessagesReceivedCount>0</RARMessagesReceivedCount>
      <RARMessagesSentCount>0</RARMessagesSentCount>
      <RARMessagesTimeoutCount>0</RARMessagesTimeoutCount>
      <RAASuccessMessagesReceivedCount>0</RAASuccessMessagesReceivedCount>
      <RAASuccessMessagesSentCount>0</RAASuccessMessagesSentCount>
      <RAAFailureMessagesReceivedCount>0</RAAFailureMessagesReceivedCount>
      <RAAFailureMessagesSentCount>0</RAAFailureMessagesSentCount>
      <CCRMessagesReceivedCount>0</CCRMessagesReceivedCount>
      <CCRMessagesSentCount>0</CCRMessagesSentCount>
      <CCRMessagesTimeoutCount>0</CCRMessagesTimeoutCount>
      <CCASuccessMessagesReceivedCount>0</CCASuccessMessagesReceivedCount>
      <CCASuccessMessagesSentCount>0</CCASuccessMessagesSentCount>
      <CCAFailureMessagesReceivedCount>0</CCAFailureMessagesReceivedCount>
      <CCAFailureMessagesSentCount>0</CCAFailureMessagesSentCount>
      <CCRIMessagesReceivedCount>0</CCRIMessagesReceivedCount>
      <CCRIMessagesSentCount>0</CCRIMessagesSentCount>
      <CCRIMessagesTimeoutCount>0</CCRIMessagesTimeoutCount>
      <CCRUMessagesReceivedCount>0</CCRUMessagesReceivedCount>
      <CCRUMessagesSentCount>0</CCRUMessagesSentCount>
      <CCRUMessagesTimeoutCount>0</CCRUMessagesTimeoutCount>
      <CCRTMessagesReceivedCount>0</CCRTMessagesReceivedCount>
      <CCRTMessagesSentCount>0</CCRTMessagesSentCount>
      <CCRTMessagesTimeoutCount>0</CCRTMessagesTimeoutCount>
      <CCAISuccessMessagesReceivedCount>0</CCAISuccessMessagesReceivedCount>
```

```
      <CCAISuccessMessagesSentCount>0</CCAISuccessMessagesSentCount>
      <CCAIFailureMessagesReceivedCount>0</CCAIFailureMessagesReceivedCount>
      <CCAIFailureMessagesSentCount>0</CCAIFailureMessagesSentCount>
      <CCAUSuccessMessagesReceivedCount>0</CCAUSuccessMessagesReceivedCount>
      <CCAUSuccessMessagesSentCount>0</CCAUSuccessMessagesSentCount>
      <CCAUFailureMessagesReceivedCount>0</CCAUFailureMessagesReceivedCount>
      <CCAUFailureMessagesSentCount>0</CCAUFailureMessagesSentCount>
      <CCATSuccessMessagesReceivedCount>0</CCATSuccessMessagesReceivedCount>
      <CCATSuccessMessagesSentCount>0</CCATSuccessMessagesSentCount>
      <CCATFailureMessagesReceivedCount>0</CCATFailureMessagesReceivedCount>
      <CCATFailureMessagesSentCount>0</CCATFailureMessagesSentCount>
      <ActiveSessionsCount>0</ActiveSessionsCount>
      <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
      </Sample>
  </DiameterTdfPeerStats>
</Statistics>
```

The MRA response follows the **Statistics ➤ DiameterShPeerStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ossiXmlOm-edit.xsd">
<DiameterShPeerStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <Name>NetworkElementName</Name>
      <NeId>NeId</NeId>
      <NetworkElementType>NetworkElementType</NetworkElementType>
      <NetworkElementSubType>NetworkElementSubType</NetworkElementSubType>
      ...
      <UDRMessagesReceivedCount>0</UDRMessagesReceivedCount>
      <UDRMessagesSentCount>0</UDRMessagesSentCount>
      <UDRMessagesTimeoutCount>0</UDRMessagesTimeoutCount>
      ...
      <PURMessagesReceivedCount>0</PURMessagesReceivedCount>
      <PURMessagesSentCount>0</PURMessagesSentCount>
      <PURMessagesTimeoutCount>0</PURMessagesTimeoutCount>
      ...
      <SNRMessagesReceivedCount>0</SNRMessagesReceivedCount>
      <SNRMessagesSentCount>0</SNRMessagesSentCount>
      <SNRMessagesTimeoutCount>0</SNRMessagesTimeoutCount>
      ...
      <MaximumActiveSessionsCount>0</MaximumActiveSessionsCount>
   </Sample>
  </DiameterShPeerStats>
</Statistics>
```

The MRA response follows the **Statistics ➤ DiameterMraDrmaStats** tag defined in the XSDs.

- **DBR Timeout**: The numeric count of Diameter Binding Request messages sent and Diameter Binding Answer messages not received in a set amount of time.
- **RUR Timeout**: The numeric count of Diameter Routing Update Request messages sent and Diameter Routing Update messages not received in a set amount of time.
- **LNR Timeout**: The numeric count of Diameter Load Notify Request messages sent and Diameter Load Notify Answer messages not received in a set amount of time.
- **LSR Timeout**:The numeric count of Diameter Load Subscription Request messages sent and Diameter Load Subscription Answer messages not received in a set amount of time.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <MRA>MRA</MRA>
      <CurrentConnectionsCount>0</CurrentConnectionsCount>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <DBRMessagesReceivedCount>0</DBRMessagesReceivedCount>
      <DBRMessageSentCount>0</DBRMessagesSentCount>
      <DBRMessageTimeoutCount>0</DBRMessagesTimeoutCount>
      ...
      <RURMessagesReceivedCount>0</RURMessagesReceivedCount>
      <RURMessagesSentCount>0</RURMessagesSentCount>
      <RURMessagesTimeoutCount>0</RURMessagesTimeoutCount>
      ...
      <LNRMessagesReceivedCount>0</LNRMessagesReceivedCount>
      <LNRMessagesSentCount>0</LNRMessagesSentCount>
      <LNRMessagesTimeoutCount>0</LNRMessagesTimeoutCount>
      ...
      <LSRMessagesReceivedCount>0</LSRMessagesReceivedCount>
      <LSRMessagesSentCount>0</LSRMessagesSentCount>
      <LSRMessagesTimeoutCount>0</LSRMessagesTimeoutCount>
      <SQRMessagesReceivedCount>0</SQRMessagesReceivedCount>
      < SQRMessagesSentCount>0</SQRMessagesSentCount>
      < SQRMessagesTimeoutCount>0</SQRMessagesTimeoutCount>
      ...
      <PeerReopenCount>0</PeerReopenCount>
    </Sample>
  </DiameterMraDrmaStats>
</Statistics>
```

The MRA response follows the **Statistics ➤ DiameterMraDrmaPeerStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
<DiameterMraDrmaPeerStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <MRA>MRA</MRA>
      <Name>NetworkElementName</Name>
      <ConnectTime>2001-11-31T12:00:00</ConnectTime>
      <DisconnectTime>2001-12-31T12:00:00</DisconnectTime>
      <ConnectAddress>10.15.25.110</ConnectAddress>
      <ConnectPort>3868</ConnectPort>
      <MessagesInCount>0</MessagesInCount>
      <MessagesOutCount>0</MessagesOutCount>
      <MessagesErrorInCount>0</MessagesErrorInCount>
      <MessagesErrorOutCount>0</MessagesErrorOutCount>
      <DBRMessagesReceivedCount>0</DBRMessagesReceivedCount>
      <DBRMessagesSentCount>0</DBRMessagesSentCount>
      <DBRMessagesTimeoutCount>0</DBRMessagesTimeoutCount>
      ...
      <RURMessagesReceivedCount>0</RURMessagesReceivedCount>
      <RURMessagesSentCount>0</RURMessagesSentCount>
      <RURMessagesTimeoutCount>0</RURMessagesTimeoutCount>
      ...
      <LNRMessagesReceivedCount>0</LNRMessagesReceivedCount>
```

```
      <LNRMessagesSentCount>0</LNRMessagesSentCount>
      <LNRMessagesTimeoutCount>0</LNRMessagesTimeoutCount>
      <LSRMessagesReceivedCount>0</LSRMessagesReceivedCount>
      <LSRMessagesSentCount>0</LSRMessagesSentCount>
      <LSRMessagesTimeoutCount>0</LSRMessagesTimeoutCount>
      <SQRMessagesReceivedCount>0</SQRMessagesReceivedCount>
      < SQRMessagesSentCount>0</SQRMessagesSentCount>
      < SQRMessagesTimeoutCount>0</SQRMessagesTimeoutCount
      ...
      <AllPcrfsDownRecvCount>0</AllPcrfsDownRecvCount>
    </Sample>
  </DiameterMraDrmaPeerStats>
</Statistics>
```

Traffic Profile Statistics

The following examples show the request and response that are defined in the XSDs for the TrafficProfileStats tag.

Request

This request follows the **QueryOmStats ➤ TrafficProfileStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2001-12-31T12:00:00</StartTime>
    <EndTime>2001-12-31T12:15:00</EndTime>
    <TrafficProfileStats>
    <PolicyServer>PolicyServer</PolicyServer>
       <Name>TrafficProfileName</Name>
    </TrafficProfileStats>
</QueryOmStats>
```

The response to this request follows the **Statistics ➤ TrafficProfileStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <TrafficProfileStats>
   <Sample>
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <Name>TrafficProfileName</Name>
      <NumberOfTimesInstalledAttemps>0</NumberOfTimesInstalledAttemps>
      <NumberOfTimesRemovedByPCRF>0</NumberOfTimesRemovedByPCRF>
     <NumberOfTimesFailedOrRemovedByGateway>0</NumberOfTimesFailedOrRemovedByGateway>

  </TrafficProfileStats>
```

Individual statistics are defined for Traffic Profiles as follows:

- **Install Attempt**: COUNTER which tracks the number of times the PCRF attempts to install a specific Traffic Profile.
- **Removed by PCRF**: COUNTER which tracks the number of times the PCRF initiates the removal of a specific Traffic Profile.

- **Failed or Removed by Gateway**: COUNTER which tracks the number of times specific Traffic Profiles fail to install on the gateway, and the number of times the gateway removes a rule without PCRF approval.

Connection Error Statistics

The following examples show the request and response that are defined in the XSDs for the ConnectionErrorStats tag.

Request

This request follows the **QueryOmStats ➤ ConnectionErrorStats** tag defined in the XSDs.

The following is an example of a request for all connection error statistics:

```
<?xml version="1.0" encoding="UTF-8"?>
<XmlInterfaceRequest>
  <QueryOmStats  DeltaCount="false">
    <StartTime>2010-04-02T00:01:00</StartTime>
    <EndTime>2011-05-21T23:59:00</EndTime>
    <ConnectionErrorStats>
      <Name>123</Name>
    </ConnectionErrorStats>
  </QueryOmStats>
</XmlInterfaceRequest>
```

Response

The response to this request follows the **Statistics ➤ ConnectionErrorStats** tag defined in the XSDs.

- DiameterCommandUnsupported - The request contained a command code that the receiver did not recognize or support.
- DiameterUnableToDeliver - This error is given when the diameter node cannot deliver the message to the destination, either because no host within the realm that supports the required application was available to process the request, or because Destination-Host AVP was given without the associated Destination-Realm AVP.
- DiameterRealmNotServed - The intended realm of the request is not recognized.
- DiameterTooBusy - The diameter node has too much traffic. When returned, a diameter node attempts to send the message to an alternate peer.
- DiameterLoopDetected - An agent detected a loop while trying to get the message to the intended recipient. The message may be sent to an alternate peer if available.
- DiameterRedirectIndication - A redirect agent has determined that the request could not be satisfied locally and the initiator of the request should direct the request directly to the server, whose contact information has been added to the response. When set, the Redirect-Host AVP must be present.
- DiameterApplicationUnsupported - A request was sent for an application that is not supported.
- DiameterInvalidHdrBits - A request was received whose bits in the diameter header were either set to an invalid combination, or to a value that is inconsistent with the command code's definition.
- DiameterInvalidAvpBits - A request was received that included an AVP whose flag bits are set to an unrecognized value, or that is inconsistent with the AVP's definition.
- DiameterUnknownPeer - A CER was received from an unknown peer.
- DiameterAuthenticationRejected - The authentication process for the user failed, most likely due to an invalid password used by the user. The user is then prompted for a new password.

- DiameterOutOfSpace - A diameter node received the accounting request but was unable to commit it to stable storage due to a temporary lack of space.
- ElectionLost - The peer has lost the election process and has disconnected the transport connection.
- DiameterEndUserServiceDenied - The credit-control server denied the service request due to service restrictions. If the CCR contained used-service-units, they are deducted, if possible.
- DiameterCreditControlNotApplicable - The credit-control server determined that the service can be granted to the end user, but that no further credit control is needed for the service (for example, the service is free of charge).
- DiameterCreditLimitReached - The credit-control server denied the service request because the end user's account could not cover the requested service. If the CCR contained used-service-units, they are deducted, if possible.
- DiameterAvpUnsupported - The peer received a message that contained an AVP that is not recognized or supported and was marked with the Mandatory bit.
- DiameterUnknownSessionId - The request contained an unknown Session-Id.
- DiameterAuthorizationRejected - A request was received for which the user could not be authorized. This error could occur if the requested service is not permitted to the user.
- DiameterInvalidAvpValue - The request contained an AVP with an invalid value in its data portion.
- DiameterMissingAvp - The request did not contain an AVP that is required by the command code definition. If this value is sent in the Result-Code AVP, a Failed-AVP AVP is included in the message.
- DiameterResourcesExceeded - A request was received that cannot be authorized because the user has already expended its allowed resources.
- DiameterContradictingAvps - The Home Diameter server has detected AVPs in the request that contradict each other, and is not willing to provide service to the user.
- DiameterAvpNotAllowed - A message was received with an AVP that cannot be present.
- DiameterAvpOccursTooManyTimes - A message was received that included an AVP that appeared more often than permitted.
- DiameterNoCommonApplication - This error is returned when a CER message is received, and there are no common applications supported between the peers.
- DiameterUnsupportedVersion - This error is returned when a request was received with an unsupported version number.
- DiameterUnableToComply - This error is returned when a request is rejected for unspecified reasons.
- DiameterInvalidBitInHeader - This error is returned when an unrecognized bit in the diameter header is set to one (1).
- DiameterInvalidAvpLength - The request contained an AVP with an invalid length.
- DiameterInvalidMessageLength - This error is returned when a request is received with an invalid message length.
- DiameterInvalidAvpBitCombo - The request contained an AVP with an invalid AVP Flags value.
- DiameterNoCommonSecurity - This error is returned when a CER message is received, but there are no common security mechanisms supported between the peers.
- DiameterUserUnknown - The specified end user is unknown in the credit-control server.
- DiameterRatingFailed - This error code is used to inform the credit-control client that the credit-control server cannot rate the service request due to insufficient rating input, an incorrect AVP combination, or an AVP or AVP value that is not recognized or supported in the rating.
- DiameterErrorInitialParameters - The initial parameters contain an error.
- RadiusSessionContyextRemoved - The residual session context has been removed.
- RadiusInvalidEapPacket - An invalid EAP Packet was detected.
- RadiusUnsupportedAttribute - The request contained an unsupported attribute.

- RadiusMissingAttribute - A request was missing a required attribute.
- RadiusNasIdMismatch - The system was unable to match the received NAS to the stored information.
- RadiusInvalidRequest - The system has received an invalid request.
- RadiusUnsupportedService - The requested service is not supported.
- RadiusUnsupportedExtension - The requested extension is not supported.
- RadiusInvalidAttributeValue - The request contains an invalid attribute value.
- RadiusAdministrativelyProhibited - The request is administratively prohibited.
- RadiusRequestNotRoutable - The request can't be routed.
- RadiusSessionNotFound - The session context cannot be found.
- RadiusSessionNotRemoveable - The session context cannot be removed.
- RadiusProxyProcessingError - An unknown proxy processing error has occurred.
- RadiusResourcesUnavailable - The necessary resources are unavailable.
- RadiusRequestInitiated - A request has been initiated.
- RadiusMultiSessionSelectionUnsupported - The requested multiple-session selection is not supported.
- RadiusLocationInfoRequired - The location information is missing.

The following is an example of the ConnectionErrorStats tag:

```
<?xml version='1.0' ?>
<Statistics>
  <ConnectionErrorStats>
    <Sample>
      <StartTime>2011-02-22T03:45:11Z</StartTime>
      <EndTime>2011-02-22T03:45:33Z</EndTime>
      <PolicyServer>rc178</PolicyServer>
      <Name>123</Name>
      <NeId></NeId>
      <DiameterCommandUnsupportedReceived>0</DiameterCommandUnsupportedReceived>
      <DiameterCommandUnsupportedSent>0</DiameterCommandUnsupportedSent>
      <DiameterUnableToDeliverReceived>0</DiameterUnableToDeliverReceived>
      <DiameterUnableToDeliverSent>0</DiameterUnableToDeliverSent>
      <DiameterRealmNotServedReceived>0</DiameterRealmNotServedReceived>
      <DiameterRealmNotServedSent>0</DiameterRealmNotServedSent>
      <DiameterTooBusyReceived>0</DiameterTooBusyReceived>
      <DiameterTooBusySent>0</DiameterTooBusySent>
      <DiameterLoopDetectedReceived>0</DiameterLoopDetectedReceived>
      <DiameterLoopDetectedSent>0</DiameterLoopDetectedSent>
      <DiameterRedirectIndicationReceived>0</DiameterRedirectIndicationReceived>
      <DiameterRedirectIndicationSent>0</DiameterRedirectIndicationSent>
      <DiameterApplicationUnsupportedReceived>0</DiameterApplication
         UnsupportedReceived>
      <DiameterApplicationUnsupportedSent>0</DiameterApplicationUnsupportedSent>
      <DiameterInvalidHdrBitsReceived>0</DiameterInvalidHdrBitsReceived>
      <DiameterInvalidHdrBitsSent>0</DiameterInvalidHdrBitsSent>
      <DiameterInvalidAvpBitsReceived>0</DiameterInvalidAvpBitsReceived>
      <DiameterInvalidAvpBitsSent>0</DiameterInvalidAvpBitsSent>
      <DiameterUnknownPeerReceived>0</DiameterUnknownPeerReceived>
      <DiameterUnknownPeerSent>0</DiameterUnknownPeerSent>
      <DiameterAuthenticationRejectedReceived>0</DiameterAuthentication
         RejectedReceived>
      <DiameterAuthenticationRejectedSent>0</DiameterAuthenticationRejectedSent>
      <DiameterOutOfSpaceReceived>0</DiameterOutOfSpaceReceived>
      <DiameterOutOfSpaceSent>0</DiameterOutOfSpaceSent>
      <ElectionLostReceived>0</ElectionLostReceived>
      <ElectionLostSent>0</ElectionLostSent>
      <DiameterEndUserServiceDeniedReceived>0</DiameterEndUserService
         DeniedReceived>
```

```
<DiameterEndUserServiceDeniedSent>0</DiameterEndUserServiceDeniedSent>
<DiameterCreditControlNotApplicableReceived>0</DiameterCreditControlNot
   ApplicableReceived>
<DiameterCreditControlNotApplicableSent>0</DiameterCreditControlNot
   ApplicableSent>
<DiameterCreditLimitReachedReceived>0</DiameterCreditLimitReachedReceived>
<DiameterCreditLimitReachedSent>0</DiameterCreditLimitReachedSent>
<DiameterAvpUnsupportedReceived>0</DiameterAvpUnsupportedReceived>
<DiameterAvpUnsupportedSent>0</DiameterAvpUnsupportedSent>
<DiameterUnknownSessionIdReceived>0</DiameterUnknownSessionIdReceived>
<DiameterUnknownSessionIdSent>0</DiameterUnknownSessionIdSent>
<DiameterAuthorizationRejectedReceived>0</DiameterAuthorization
   RejectedReceived>
<DiameterAuthorizationRejectedSent>0</DiameterAuthorizationRejectedSent>
<DiameterInvalidAvpValueReceived>0</DiameterInvalidAvpValueReceived>
<DiameterInvalidAvpValueSent>0</DiameterInvalidAvpValueSent>
<DiameterMissingAvpReceived>0</DiameterMissingAvpReceived>
<DiameterMissingAvpSent>0</DiameterMissingAvpSent>
<DiameterResourcesExceededReceived>0</DiameterResourcesExceededReceived>
<DiameterResourcesExceededSent>0</DiameterResourcesExceededSent>
<DiameterContradictingAvpsReceived>0</DiameterContradictingAvpsReceived>
<DiameterContradictingAvpsSent>0</DiameterContradictingAvpsSent>
<DiameterAvpNotAllowedReceived>0</DiameterAvpNotAllowedReceived>
<DiameterAvpNotAllowedSent>0</DiameterAvpNotAllowedSent>
<DiameterAvpOccursTooManyTimesReceived>0</DiameterAvpOccursToo
   ManyTimesReceived>
<DiameterAvpOccursTooManyTimesSent>0</DiameterAvpOccursTooManyTimesSent>
<DiameterNoCommonApplicationReceived>0</DiameterNoCommon
   ApplicationReceived>
<DiameterNoCommonApplicationSent>0</DiameterNoCommonApplicationSent>
<DiameterUnsupportedVersionReceived>0</DiameterUnsupportedVersionReceived>
<DiameterUnsupportedVersionSent>0</DiameterUnsupportedVersionSent>
<DiameterUnableToComplyReceived>0</DiameterUnableToComplyReceived>
<DiameterUnableToComplySent>0</DiameterUnableToComplySent>
<DiameterInvalidBitInHeaderReceived>0</DiameterInvalidBitInHeaderReceived>
<DiameterInvalidBitInHeaderSent>0</DiameterInvalidBitInHeaderSent>
<DiameterInvalidAvpLengthReceived>0</DiameterInvalidAvpLengthReceived>
<DiameterInvalidAvpLengthSent>0</DiameterInvalidAvpLengthSent>
<DiameterInvalidMessageLengthReceived>0</DiameterInvalidMessage
   LengthReceived>
<DiameterInvalidMessageLengthSent>0</DiameterInvalidMessageLengthSent>
<DiameterInvalidAvpBitComboReceived>0</DiameterInvalidAvpBitComboReceived>
<DiameterInvalidAvpBitComboSent>0</DiameterInvalidAvpBitComboSent>
<DiameterNoCommonSecurityReceived>0</DiameterNoCommonSecurityReceived>
<DiameterNoCommonSecuritySent>0</DiameterNoCommonSecuritySent>
<DiameterUserUnknownReceived>0</DiameterUserUnknownReceived>
<DiameterUserUnknownSent>0</DiameterUserUnknownSent>
<DiameterRatingFailedReceived>0</DiameterRatingFailedReceived>
<DiameterRatingFailedSent>0</DiameterRatingFailedSent>
<DiameterErrorInitialParametersReceived>0</DiameterErrorInitial
   ParametersReceived>
<DiameterErrorInitialParametersSent>2</DiameterErrorInitialParametersSent>
<RadiusSessionContyextRemovedReceived>0</RadiusSessionContyext
   RemovedReceived>
<RadiusSessionContyextRemovedSent>0</RadiusSessionContyextRemovedSent>
<RadiusInvalidEapPacketReceived>0</RadiusInvalidEapPacketReceived>
<RadiusInvalidEapPacketSent>0</RadiusInvalidEapPacketSent>
<RadiusUnsupportedAttributeReceived>0</RadiusUnsupportedAttributeReceived>
<RadiusUnsupportedAttributeSent>0</RadiusUnsupportedAttributeSent>
<RadiusMissingAttributeReceived>0</RadiusMissingAttributeReceived>
<RadiusMissingAttributeSent>0</RadiusMissingAttributeSent>
<RadiusNasIdMismatchReceived>0</RadiusNasIdMismatchReceived>
<RadiusNasIdMismatchSent>0</RadiusNasIdMismatchSent>
<RadiusInvalidRequestReceived>0</RadiusInvalidRequestReceived>
<RadiusInvalidRequestSent>0</RadiusInvalidRequestSent>
```

```
      <RadiusUnsupportedServiceReceived>0</RadiusUnsupportedServiceReceived>
      <RadiusUnsupportedServiceSent>0</RadiusUnsupportedServiceSent>
      <RadiusUnsupportedExtensionReceived>0</RadiusUnsupportedExtensionReceived>
      <RadiusUnsupportedExtensionSent>0</RadiusUnsupportedExtensionSent>
      <RadiusInvalidAttributeValueReceived>0</RadiusInvalidAttributeValueReceived>
      <RadiusInvalidAttributeValueSent>0</RadiusInvalidAttributeValueSent>
      <RadiusAdministrativelyProhibitedReceived>0</RadiusAdministratively
         ProhibitedReceived>
      <RadiusAdministrativelyProhibitedSent>0</RadiusAdministratively
         ProhibitedSent>
      <RadiusRequestNotRoutableReceived>0</RadiusRequestNotRoutableReceived>
      <RadiusRequestNotRoutableSent>0</RadiusRequestNotRoutableSent>
      <RadiusSessionNotFoundReceived>0</RadiusSessionNotFoundReceived>
      <RadiusSessionNotFoundSent>0</RadiusSessionNotFoundSent>
      <RadiusSessionNotRemoveableReceived>0</RadiusSessionNotRemoveableReceived>
      <RadiusSessionNotRemoveableSent>0</RadiusSessionNotRemoveableSent>
      <RadiusProxyProcessingErrorReceived>0</RadiusProxyProcessingErrorReceived>
      <RadiusProxyProcessingErrorSent>0</RadiusProxyProcessingErrorSent>
      <RadiusResourcesUnavailableReceived>0</RadiusResourcesUnavailableReceived>
      <RadiusResourcesUnavailableSent>0</RadiusResourcesUnavailableSent>
      <RadiusRequestInitiatedReceived>0</RadiusRequestInitiatedReceived>
      <RadiusRequestInitiatedSent>0</RadiusRequestInitiatedSent>
      <RadiusMultiSessionSelectionUnsupportedReceived>0</RadiusMultiSession
         SelectionUnsupportedReceived>
      <RadiusMultiSessionSelectionUnsupportedSent>0</RadiusMultiSessionSelection
         UnsupportedSent>
      <RadiusLocationInfoRequiredReceived>0</RadiusLocationInfoRequiredReceived>
      <RadiusLocationInfoRequiredSent>0</RadiusLocationInfoRequiredSent>
    </Sample>
  </ConnectionErrorStats>
</Statistics>
```

Quota Profile Statistics

The following examples show the request and response that are defined in the XSDs for the QuotaProfileStats tag.

Request

This request follows the **QueryOmStats ➤ QuotaProfileStats** tag defined in the XSDs.

The following is an example of a request for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryOmStats>
    <StartTime>2001-12-31T12:00:00</StartTime>
    <EndTime>2001-12-31T12:15:00</EndTime>
    <QuotaProfileStats>
      <PolicyServer>PolicyServer</PolicyServer>
    </QuotaProfileStats>
</QueryOmStats>
```

The response to this request follows the **Statistics ➤ QuotaProfileStats** tag defined in the XSDs.

The following is an example of a response for a single policy server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Statistics>
  <QuotaProfileStats>
   <Sample>
```

```
      <StartTime>2001-12-31T12:00:00</StartTime>
      <EndTime>2001-12-31T12:15:00</EndTime>
      <PolicyServer>PolicyServer</PolicyServer>
      <Name>QuotaProfileName</Name>
      <NumberOfTimesActivated>0</NumberOfTimesActivated>
      <NumberOfTimesVolumnThresholdReached>0</NumberOfTimesVolumnThresholdReached>
      <NumberOfTimesTimeThresholdReached>0</NumberOfTimesTimeThresholdReached>
      <NumberOfTimesEventThresholdReached>0</NumberOfTimesEventThresholdReached>
   </QuotaProfileStats>
```

Individual statistics are defined as follows for Quota Profiles:

- **Activated**: COUNTER which tracks the number of times that a specific Quota Profile is activated.
- **Volume Threshold Reached**: COUNTER which tracks the number of times the configured volume threshold limit is reached for a specific Quota Profile.
- **Time Threshold Reached**: COUNTER which tracks the number of times the configured time threshold limit is reached for a specific Quota Profile.
- **Event Threshold Reached**: COUNTER which tracks the number of times the configured event threshold limit is reached for a specific Quota Profile.

Interval Statistics/PCRF Session License Tracking and Reporting

The following XML requests and responses apply to interval statistics and PCRF session license tracking and reporting. The Interval Stats tag allows for the collection of specific maximum values for some counters over a time interval of 15 minutes. By using the OSSI/data collection feature maximums can be collected for each 15-minute interval of a day, over a 30-day period, to determine peak usage times.

The following examples show the request and response that are defined in the XSDs for the IntervalStats tag.

MPE Request

This request follows the **QueryOmStats ➤ IntervalStats** tag defined in the XSDs.

The following is an example of this request for an MPE.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <QueryOmStats>
    <StartTime>2001-08-01T10:00:01Z</StartTime>
    <IntervalStats>
        <PolicyServer>name</PolicyServer> (policy server name, if this field does
not exist, all policy servers are included in response)
    </IntervalStats>
  </QueryOmStats>
</XmlInterfaceRequest>
```

Where the TIMESTAMP: is a time stamp in the form:

YYYY-MM-DD- ***T*** HH:MM:SS ***Z***

MPE Response

The response to this MPE request follows the **Statistics ➤ IntervalStats** tag defined in the XSDs.

Where:

- IntervalStartTime - The time when this sub-system began collecting the data presented. If this is the first instance of the sub-system collecting data, this value may be N/A.
- ConfiguredLength - This value is always 900 seconds, which is 15 minutes.
- ActualLength - The length of the interval. Normally, this is the same value as ConfiguredLength, however there are two cases where these values will differ:

  1. Cluster has just started and no data is available (ActualLength = 0)
  2. Cluster has started and an interval completed but the software did not start on a quarter boundary (e.g. ActualLength = nnn where nnn is any number between 0 and 900. If the software started 200 seconds after the 15-min boundary, the when the interval is completed, the ActualLength = 700).

- IsComplete - If the ConfiguredLength has completed, then the value is true. If not, then the value is false.
- IntervalMaxTransactionsPerSecond - The maximum value of the statistic MaxTransactionsPerSecond for the interval reported. As with MaxTransactionsPerSecond, this count does not include SH traffic.
- IntervalMaxMRABindingCount - The maximum value of the statistic MaxMRABindingCount for the interval reported. This value is 0 on MPEs.
- IntervalMaxSessionCount - The maximum value of the statistic MaxSessionCount for the interval reported.
- IntervalMaxPDNConnectionCount - The maximum value of the statistic MaxPDNConnectionCount for the interval reported.

For example:

```
<?xml version='1.0' ?>
<Statistics>
  <IntervalStats>
    <Sample>
      <StartTime>2011-05-04T13:23:52Z</StartTime>
      <EndTime>2011-05-04T13:23:52Z</EndTime>
      <PolicyServer>testmpe</PolicyServer>
      <IntervalStartTime>2001-05-04T13:15:00Z</IntervalStartTime>
      <ConfiguredLength>900</ConfiguredLength>
      <ActualLength>12</ActualLength>
      <IsComplete>false</IsComplete>
      <IntervalMaxTransactionsPerSecond>5</IntervalMaxTransactionsPerSecond>
      <IntervalMaxMRABindingCount>16</IntervalMaxMRABindingCount>
      <IntervalMaxSessionCount>12</IntervalMaxSessionCount>
      <IntervalMaxPDNConnectionCount>12</IntervalMaxPDNConnectionCount>
    </Sample>
    <Sample>
      <StartTime>2011-05-04T13:23:52Z</StartTime>
      <EndTime>2011-05-06T05:57:36Z</EndTime>
      <PolicyServer>testmpe</PolicyServer>
      <IntervalStartTime>2001-08-01T10:00:01Z</IntervalStartTime>
      <ConfiguredLength>900</ConfiguredLength>
      <ActualLength>12</ActualLength>
      <IsComplete>false</IsComplete>
      <IntervalMaxTransactionsPerSecond>5</IntervalMaxTransactionsPerSecond>
      <IntervalMaxMRABindingCount>16</IntervalMaxMRABindingCount>
      <IntervalMaxSessionCount>12</IntervalMaxSessionCount>
      <IntervalMaxPDNConnectionCount>12</IntervalMaxPDNConnectionCount>
    </Sample>
  </IntervalStats>
</Statistics>
```

MRA Request

This request follows the **QueryOmStats ➤ IntervalStats** tag defined in the XSDs.

The following is an example of this request for an MRA.

Where:

- IntervalStartTime - The time when this sub-system began collecting the data presented. If this is the first instance of the sub-system collecting data, this value may be N/A.
- ConfiguredLength - This value is always 900 seconds, which is 15 minutes.
- ActualLength - The length of the interval. Normally, this is the same value as ConfiguredLength, however there are two cases (documented above) where these values will differ:

  1. Cluster has just started and no data is available (ActualLength = 0)
  2. Cluster has started and an interval completed but the software did not start on a quarter boundary (e.g. ActualLength = nnn where nnn is any number between 0 and 900. If the software started 200 seconds after the 15-min boundary, the when the interval is completed, the ActualLength = 700).

- IsComplete - If the ConfiguredLength has completed, then the value is true. If not, then the value is false.
- IntervalMaxTransactionsPerSecond - The maximum value of the statistic MaxTransactionsPerSecond for the interval reported. As with MaxTransactionsPerSecond, this count does not include SH traffic.
- IntervalMaxMRABindingCount - The maximum value of the statistic MaxMRABindingCount for the interval reported.
- IntervalMaxSessionCount - The maximum value of the statistic MaxSessionCount for the interval reported.
- IntervalMaxPDNConnectionCount - The maximum value of the statistic MaxPDNConnectionCount for the interval reported.

For example:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <QueryOmStats>
    <StartTime>2001-08-01T10:00:01Z</StartTime>
    <IntervalMraStats>
        <MRA>testmra</MRA>
    </IntervalMraStats>
  </QueryOmStats>
</XmlInterfaceRequest>
```

MRA Response

The response to this MRA request follows the **Statistics ➤ IntervalStats** tag defined in the XSDs. For example:

```
<?xml version='1.0' ?>
<Statistics>
  <IntervalMraStats>
    <Sample>
      <StartTime>2011-05-04T13:23:52Z</StartTime>
      <EndTime>2011-05-06T08:15:26Z</EndTime>
      <MRA>testmra</MRA>
      <IntervalStartTime>2011-04-04 09:23:52.0</IntervalStartTime>
      <ConfiguredLength>900</ConfiguredLength>
      <ActualLength>12</ActualLength>
      <IsComplete>false</IsComplete>
      <IntervalMaxTransactionsPerSecond>5</IntervalMaxTransactionsPerSecond>
```

```
      <IntervalMaxMRABindingCount>16</IntervalMaxMRABindingCount>
      <IntervalMaxSessionCount>12</IntervalMaxSessionCount>
      <IntervalMaxPDNConnectionCount>12</IntervalMaxPDNConnectionCount>
    </Sample>
  </IntervalMraStats>
</Statistics>
```

Chapter 8

Identity Management (IDM)

Topics:

Identity Management (IDM) allows the CMP to configure user names, passwords and roles using the OSSI interface. It also allows querying of user scopes and roles.

Identity Management Overview

The Identity Management system connects to a CMP through an HTTP or HTTPS interface. The user must specify the operation parameter **idm** in the requested URL, similar to `http://127.0.0.1/mi/xmlInterfaceRequest.do?user=supervisor&pwd=password&operation=idm`.

The account used for connecting the OSSI to the CMP is a regular CMP user account, with the default global scope and the following privileges:

- Show Privilege for OSSI Import/Export
- Read-Write Privilege for User Management

The Identity Management feature allows users to perform the following tasks:

- Add a User - Create one or more users, or update a current user's parameters by overwriting them with new parameters.
- Update a User - Update parameters for one or more users.
- Query a User - Query one user if the Name element occurs, or all users if no Name element occurs.
- Delete a User - Delete one or more users.
- Query Role - Query one role if the Name element occurs, or all roles if no Name element occurs.
- Query Scope - Query one scope if the Name element occurs, or all scopes if no Name element occurs.

Add a User

The following examples show the request and response that are defined in the XSDs for the AddSysAdminUser tag.

Successful Request

Request

This request follows the AddSysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <AddSysAdminUser>
    <SysAdminUser>
      <Name>admin1</Name>
     <Description>The default administrator user with all privileges</Description>

      <Password>camiant</Password>
      <LockedStatus>0</LockedStatus>
      <RoleRef>
        <Name>Administrator</Name>
      </RoleRef>
      <ScopeRef>
        <Name>Global</Name>
      </ScopeRef>
    </SysAdminUser>
```

```
  </AddSysAdminUser>
</XmlInterfaceRequest>
```

Table 3: AddSysAdminUser Operation Parameters

Parameter Name	Optional/Required	Limitation
User Name	Required	250 characters (string)
Description	Optional	250 characters (string)
Password	Required	A clear text
LockedStatus	Required	0 = unlocked; 1 = locked
RoleRef	Required	Can specify only one role
ScopeRef	Required	Can specify multiple scopes

Response

This request follows the AddSysAdminUser tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
  <Result>0</Result>
  <Command type="XmlInterfaceResponse">
  <Success count="1">Successfully imported 1 User(s).</Success>
  <Failure count="0"></Failure>
 </Command>
</Response>
```

The errors that a user may receive are shown below.

Table 4: AddSysAdminUser Operation Errors

Error	Message	Description
ERROR_USER_INVALID_NAME	Invalid User Name for: '{$UserName}'	UserName is an empty string.
ERROR_USER_NAME_TOO_LONG	User Name exceeds max length for: '{$UserName}'	UserName exceeds max length (250 characters).
ERROR_USER_DESCRIPTION_TOO_LONG	User Description exceeds max length. User: '{$UserName}'	Description exceeds max length (250 characters)

Error	Message	Description
ERROR_USER_NOROLE_OR_NOSCOPE	User: '{$Username}' must have an associated Role and Scope.	Non-Role or non-Scope is defined in OSSI command.
ERROR_USER_INVALID_ROLE_NAME	Invalid Role Name for: '{$RoleName}'	RoleName is an empty string, or RoleName exceeds max length (250 characters), or There is more than one Role defined.
ERROR_USER_INVALID_SCOPE_NAME	Invalid Scope Name for: '{$ScopeName}'	ScopeName is an empty string, or ScopeName is exceeds max length (250 characters).
ERROR_USER_ROLE_LINK	The user: {$UserName}'s associated role does not exist. Please create it first.	RoleName is not defined in CMP.
ERROR_USER_SCOPE_LINK	The user: {$UserName}'s associated scope does not exist. Please create it first.	ScopeName is not defined in CMP.
ERROR_SCHEMA_INVALID	N/A	If the input OSSI command cannot match the schema specification, there will be an error-message response to the operator. For example, if you define a LockStatus with a string, the following message will be reported: `“Error parsing import file: Error parsing import file: cvc-datatype-valid.1.2.1: 'ABC' is not a valid value for 'integer'.”`

ERROR_USER_INVALID_NAME

Request

This request follows the AddSysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
```

```
  <AddSysAdminUser>
    <SysAdminUser>
      <Name></Name>
     <Description>The default administrator user with all privileges</Description>

      <Password>tekelec</Password>
      <LockedStatus>0</LockedStatus>
      <RoleRef>
        <Name>Administrator</Name>
      </RoleRef>
      <ScopeRef>
        <Name>Global</Name>
      </ScopeRef>
    </SysAdminUser>
  </AddSysAdminUser>
</XmlInterfaceRequest>
```

Response

The following error response occurs when the user name is invalid.

```
<?xml version='1.0' ?>
<Response>
  <Result>0</Result>
   <Command type="XmlInterfaceResponse">
   <Success count="0"></Success>
   <Failure count="1">Failed to import 1 User(s).

   Invalid User Name for: ''</Failure>
  </Command>
</Response>
```

ERROR_USER_INVALID_ROLE_NAME

Request

This request follows the AddSysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <AddSysAdminUser>
    <SysAdminUser>
      <Name>admin1</Name>
     <Description>The default administrator user with all privileges</Description>

      <Password>tekelec</Password>
      <LockedStatus>0</LockedStatus>
      <RoleRef>
        <Name></Name>
      </RoleRef>
      <ScopeRef>
        <Name>Global</Name>
      </ScopeRef>
    </SysAdminUser>
  </AddSysAdminUser>
</XmlInterfaceRequest>
```

Response

The following error response occurs when the role name is invalid.

```
<?xml version='1.0' ?>
<Response>
```

```
 <Result>0</Result>
  <Command type="XmlInterfaceResponse">
  <Success count="0"></Success>
  <Failure count="1">Failed to import 1 User(s).

  Invalid Role Name for: 'admin1'</Failure>
</Command>
</Response>
```

ERROR_USER_INVALID_SCOPE_NAME

Request

This request follows the AddSysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <AddSysAdminUser>
    <SysAdminUser>
      <Name>admin1</Name>
     <Description>The default administrator user with all privileges</Description>

      <Password>tekelec</Password>
      <LockedStatus>0</LockedStatus>
      <RoleRef>
        <Name>Administrator</Name>
      </RoleRef>
      <ScopeRef>
        <Name></Name>
      </ScopeRef>
    </SysAdminUser>
  </AddSysAdminUser>
</XmlInterfaceRequest>
```

Response

The following error response occurs when the scope name is invalid.

```
<?xml version='1.0' ?>
<Response>
  <Result>0</Result>
  <Command type="XmlInterfaceResponse">
  <Success count="0"></Success>
  <Failure count="1">Failed to import 1 User(s).

  Invalid Scope Name for: 'admin1'</Failure>
</Command>
</Response>
```

ERROR_USER_ROLE_LINK

Request

This request follows the AddSysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <AddSysAdminUser>
    <SysAdminUser>
      <Name>admin1</Name>
     <Description>The default administrator user with all privileges</Description>
```

```
      <Password>tekelec</Password>
      <LockedStatus>0</LockedStatus>
      <ScopeRef>
        <Name>Not Exist Global</Name>
      </ScopeRef>
    </SysAdminUser>
  </AddSysAdminUser>
</XmlInterfaceRequest>
```

Response

The following error response occurs when the user's associated role does not exist.

```
<?xml version='1.0' ?>
<Response>
  <Result>0</Result>
  <Command type="XmlInterfaceResponse">
  <Success count="0"></Success>
  <Failure count="1">Failed to import 1 User(s).

  User: 'admin1' must have an associated Role and Scope.</Failure>
</Command>
</Response>
```

ERROR_USER_NOROLE_OR_NOSCOPE

Request

This request follows the AddSysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <AddSysAdminUser>
    <SysAdminUser>
      <Name>admin1</Name>
     <Description>The default administrator user with all privileges</Description>

      <Password>tekelec</Password>
      <LockedStatus>0</LockedStatus>
      <RoleRef>
        <Name>Not Exits Administrator</Name>
      </RoleRef>
      <ScopeRef>
        <Name>Global</Name>
      </ScopeRef>
    </SysAdminUser>
  </AddSysAdminUser>
</XmlInterfaceRequest>
```

Response

The following error response occurs when the user's role does not exist.

```
<?xml version='1.0' ?>
<Response>
  <Result>0</Result>
  <Command type="XmlInterfaceResponse">
  <Success count="0"></Success>
  <Failure count="1">Failed to import 1 User(s).

 The user: admin1's associated role does not exist. Please create it first.</Failure>
```

```
</Command>
</Response>
```

Request

This request follows the AddSysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <AddSysAdminUser>
    <SysAdminUser>
      <Name>admin1</Name>
     <Description>The default administrator user with all privileges</Description>

      <Password>tekelec</Password>
      <LockedStatus>0</LockedStatus>
      <RoleRef>
        <Name>Administrator</Name>
      </RoleRef>
      <ScopeRef>
        <Name>Not Exist Global</Name>
      </ScopeRef>
    </SysAdminUser>
  </AddSysAdminUser>
</XmlInterfaceRequest>
```

Response

The following error response occurs when the user's scope does not exist.

```
<?xml version='1.0' ?>
<Response>
  <Result>0</Result>
  <Command type="XmlInterfaceResponse">
  <Success count="0"></Success>
  <Failure count="1">Failed to import 1 User(s).

  The user: admin1's associated scope does not exist. Please create it
first.</Failure>
</Command>
</Response>
```

Update a User

The following examples show the request and response that are defined in the XSDs for the UpdateSysAdminUser tag.

Request

This request follows the UpdateSysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <UpdateSysAdminUser>
    <SysAdminUser>
      <Name>admin1</Name>
      <Description>New The default administrator user with all
         privileges</Description>
      <LockedStatus>0</LockedStatus>
      <RoleRef>
```

```
        <Name>Administrator</Name>
      </RoleRef>
      <ScopeRef>
        <Name>Global</Name>
      </ScopeRef>
    </SysAdminUser>
  </UpdateSysAdminUser>
</XmlInterfaceRequest>
```

Table 5: UpdateSysAdminUser Operation Parameters

Parameter Name	Optional/Required	Limitation
User Name	Required	250 characters (string)
Description	Optional	250 characters (string)
Password	Optional	A clear text
LockedStatus	Optional	0 = unlocked; 1 = locked
RoleRef	Optional	Can specify only one role
ScopeRef	Optional	Can specify multiple scopes

The probable errors a user may receive are shown below.

Table 6: UpdateSysAdminUser Operation Errors

Error	Message	Description
ERROR_USER_NAME_NOT_EXIST	Specified User does not exist: '{$UserName}'	The specified user is not in system.
ERROR_USER_INVALID_NAME	Invalid User Name for: '{$UserName}'	UserName is an empty string.
ERROR_USER_NAME_TOO_LONG	User Name exceeds max length for: '{$UserName}'	UserName exceeds max length (250 characters).
ERROR_USER_DESCRIPTION_TOO_LONG	User Description exceeds max length. User: '{$UserName}'	Description exceeds max length (250 characters)
ERROR_USER_NOROLE_OR_NOSCOPE	User: '{$Username}' must have an associated Role and Scope.	Non-Role or non-Scope is defined in OSSI command.

Error	Message	Description
ERROR_USER_INVALID_ROLE_NAME	Invalid Role Name for: '{$RoleName}'	RoleName exceeds max length (250 characters), or There is more than one Role defined.
ERROR_USER_INVALID_SCOPE_NAME	Invalid Scope Name for: '{$ScopeName}'	ScopeName exceeds max length (250 characters).
ERROR_USER_ROLE_LINK	The user: {$UserName}'s associated role does not exist. Please create it first.	RoleName is an empty string, or RoleName is not defined in CMP.
ERROR_USER_SCOPE_LINK	The user: {$UserName}'s associated scope does not exist. Please create it first.	ScopeName is an empty string, or ScopeName is not defined in CMP.
ERROR_SCHEMA_INVALID	N/A	If the input OSSI command cannot match the schema specification, there will be an error-message response to the operator. For example, if you define a LockStatus with a string, the following message will be reported: `“Error parsing import file: Error parsing import file: cvc-datatype-valid.1.2.1: 'ABC' is not a valid value for 'integer'.”`

Response

The response to this request follows the generic Response tag defined in the XSDs.

```
<?xml version='1.0' ?>
<Response>
 <Result>0</Result>
  <Command type="XmlInterfaceResponse">
  <Success count="1">Successfully updated 1 User(s).</Success>
  <Failure count="0"></Failure>
</Command>
</Response>
```

ERROR_USER_NAME_NOT_EXIST

Request

This request follows the UpdateSysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <UpdateSysAdminUser>
    <SysAdminUser>
      <Name>NotExistadmin1</Name>
      <Description>New The default administrator user with all
         privileges</Description>
      <LockedStatus>0</LockedStatus>
      <RoleRef>
        <Name>Administrator</Name>
      </RoleRef>
      <ScopeRef>
        <Name>Global</Name>
      </ScopeRef>
    </SysAdminUser>
  </UpdateSysAdminUser>
</XmlInterfaceRequest>
```

Response

The following error response occurs when the user name does not exist.

```
<?xml version='1.0' ?>
<Response>
  <Result>0</Result>
   <Command type="XmlInterfaceResponse">
   <Success count="0"></Success>
   <Failure count="1">Failed to update 1 User(s).

   Specified User does not exist: 'NotExistadmin1'</Failure>
 </Command>
</Response>
```

ERROR_USER_NOROLE_OR_NOSCOPE

Request

This request follows the UpdateSysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <UpdateSysAdminUser>
    <SysAdminUser>
      <Name>admin1</Name>
      <Description>New The default administrator user with all
         privileges</Description>
      <LockedStatus>0</LockedStatus>
      <RoleRef>
        <Name>not exist Administrator</Name>
      </RoleRef>
      <ScopeRef>
        <Name>Global</Name>
      </ScopeRef>
    </SysAdminUser>
```

```
  </UpdateSysAdminUser>
</XmlInterfaceRequest>
```

Response

The following error response occurs when the user's specified role does not exist.

```
<?xml version='1.0' ?>
<Response>
 <Result>0</Result>
   <Command type="XmlInterfaceResponse">
   <Success count="0"></Success>
   <Failure count="1">Failed to update 1 User(s).

   The user: admin1's associated role does not exist. Please create it
first.</Failure>
 </Command>
</Response>
```

Request

This request follows the UpdateSysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <UpdateSysAdminUser>
    <SysAdminUser>
      <Name>admin1</Name>
      <Description>New The default administrator user with all
         privileges</Description>
      <LockedStatus>0</LockedStatus>
      <RoleRef>
        <Name>Administrator</Name>
      </RoleRef>
      <ScopeRef>
        <Name>Not exist Global</Name>
      </ScopeRef>
    </SysAdminUser>
  </UpdateSysAdminUser>
</XmlInterfaceRequest>
```

Response

The following error response occurs when the user's specified scope does not exist.

```
<?xml version='1.0' ?>
<Response>
 <Result>0</Result>
   <Command type="XmlInterfaceResponse">
   <Success count="0"></Success>
   <Failure count="1">Failed to update 1 User(s).

   The user: admin1's associated scope does not exist. Please create it
first.</Failure>
 </Command>
</Response>
```

ERROR_USER_INVALID_ROLE_NAME

Request

This request follows the UpdateSysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <UpdateSysAdminUser>
    <SysAdminUser>
      <Name>admin1</Name>
      <Description>New The default administrator user with all
         privileges</Description>
      <LockedStatus>0</LockedStatus>
      <RoleRef>
        <Name>123</Name>
      </RoleRef>
      <RoleRef>
        <Name>Administrator</Name>
      </RoleRef>
      <ScopeRef>
        <Name>123</Name>
      </ScopeRef>
    </SysAdminUser>
  </UpdateSysAdminUser>
</XmlInterfaceRequest>
```

Response

The following error response occurs when the role name either exceeds the maximum length of 250 characters, or there is more than one Role defined.

```
<?xml version='1.0' ?>
 <Response>
   <Result>0</Result>
   <Command type="XmlInterfaceResponse">
   <Success count="0"></Success>
   <Failure count="1">Failed to update 1 User(s).

   Invalid Role Name for: 'admin1'</Failure>
 </Command>
</Response>
```

Query a User

The following examples show the request and response that are defined in the XSDs for the QuerySysAdminUser tag. If the Name element occurs, the CMP will query only the corresponding user; otherwise, all users are queried.

Request

This request follows the QuerySysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <QuerySysAdminUser>
      <Name>admin</Name>
```

```
    </QuerySysAdminUser>
</XmlInterfaceRequest>
```

Table 7: QuerySysAdminUser Operation Parameters

Parameter Name	Optional/Required	Limitation
Name	Optional	250 characters (string)

If the query is successful, the list of users will be returned in XML as a response with the following fields for each user.

Table 8: Output Fields for QuerySysAdminUser Operation

Field Name	Description
Name	The user's name
Description	The user's description
Password	The user's password, encrypted
LockedStatus	0 = locked, 1 = unlocked
RoleRef	The associated role
ScopeRef	The associated scope; there can be multiple scopes.

Response

The response to this request:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="7.5.0">
    <SysAdminUser>
        <Name>admin</Name>
       <Description>The default administrator user with all privileges</Description>

        <Password>835154D6D3FA2C3575AA700A54AB9F6492E7ABB7</Password>
        <LockedStatus>0</LockedStatus>
        <RoleRef>
            <Name>Administrator</Name>
        </RoleRef>
        <ScopeRef>
            <Name>Global</Name>
        </ScopeRef>
```

```
    </SysAdminUser>
</ConfigurationData>
```

Delete a User

The following examples show the request and response that are defined in the XSDs for the DeleteSysAdminUser tag.

Successful Deletion

Request

This request follows the DeleteSysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <DeleteSysAdminUser>
    <SysAdminUser>
      <Name>admin1</Name>
    </SysAdminUser>
  </DeleteSysAdminUser>
</XmlInterfaceRequest>
```

Table 9: DeleteSysAdminUser Operation Parameters

Parameter Name	Optional/Required	Limitation
Name	Required	250 characters (string)

The probable errors a user may receive are shown below.

Table 10: DeleteSysAdminUser Operation Errors

Error	Message	Description
ERROR_DELETE_FAILURE	Error deleting User Name: {$UserName}	The specified user does not exist, or "Admin" user is specified.

Response

The response to this request:

```
<?xml version='1.0' ?>
<Response>
<Result>0</Result>
<Command type="XmlInterfaceResponse">
<Success count="1">Deleted 1 users.</Success>
<Failure count="0"></Failure>
```

```
</Command>
</Response>
```

If deletion is successful, the number of deleted users will be reported.

ERROR_DELETE_FAILURE

Request

This request follows the DeleteSysAdminUser tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
  <DeleteSysAdminUser>
    <SysAdminUser>
      <Name>admin1</Name>
    </SysAdminUser>
  </DeleteSysAdminUser>
</XmlInterfaceRequest>
```

Response

The response to this request when the specified user name does not exist:

```
<?xml version='1.0' ?>
<Response>
<Result>0</Result>
<Command type="XmlInterfaceResponse">
<Success count="0"></Success>
<Failure count="1">Failed to delete 1 user.
Error deleting User Name: admin1</Failure>
</Command>
</Response>
```

Query User Role

The following examples show the request and response that are defined in the XSDs for the QueryRole tag. If the Name element occurs, the CMP will query the corresponding role with the name; otherwise, it will query all roles.

Request

This request follows the QueryRole tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <QueryRole>
      <Name>Administrator</Name>
```

```
    </QueryRole>
</XmlInterfaceRequest>
```

Table 11: QueryRole Operation Parameters

Parameter Name	Optional/Required	Limitation
Name	Optional	250 characters (string)

If the query is successful, the list of users will be returned in XML as a response with the following fields for each user.

Table 12: Output Fields for QueryRole Operation

Field Name	Description
Name	The role's name
Description	The role's description
RolePrivilege	The privilege with privilege name and access level assigned to the Role. There can be multiple privileges.

Response

The response to this request:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="7.5.0">
    <Role>
        <Name>Administrator</Name>
        <Description>The default role with all privileges</Description>
        <RolePrivilege>
            <Privilege>policy server</Privilege>
            <Level>40</Level>
        <RolePrivilege>
            <Privilege>Network Element</Privilege>
            <Level>Read-Write</Level>
        </RolePrivilege>
        <RolePrivilege>
          <Privilege>Application</Privilege>
          <Level>Read-Write</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>Policy Library</Privilege>
            <Level>Read, Deploy, and Write</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>Template Library</Privilege>
            <Level>Read-Write</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>User Management</Privilege>
```

```
            <Level>Read-Write</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>Manager Event and Audit Logs</Privilege>
            <Level>Read-Write</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>Tasks</Privilege>
            <Level>Read-Write</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>License Management</Privilege>
            <Level>Read-Write</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>Traffic Profiles</Privilege>
            <Level>Read-Write</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>XML Import / Export</Privilege>
            <Level>Show</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>Operational Measurements</Privilege>
            <Level>Read-Only</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>Charging Server</Privilege>
            <Level>Read-Write</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>Time Period</Privilege>
            <Level>Read-Write</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>Policy Import / Export</Privilege>
            <Level>Read-Write</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>Alarms</Privilege>
            <Level>Read-Write</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>Audit Log User Info</Privilege>
            <Level>Show</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>Password Strength</Privilege>
            <Level>Read-Write</Level>
        </RolePrivilege>
        <RolePrivilege>
            <Privilege>Event Log</Privilege>
            <Level>Read-Write</Level>
        </RolePrivilege>
        <RolePrivilege>
           <Privilege>Audit Log</Privilege>
           <Level>Read-Write</Level>
        </RolePrivilege>
  </Role>
</ConfigurationData>
```

A subset of the RolePrivilege element will be outputted in XML based on the mode selected by the operator. The output XML cannot be imported to the CMP again.

Query User Scope

The following examples show the request and response that are defined in the XSDs for the QueryScope tag. If the Name element occurs, the CMP will query only the corresponding scope, otherwise all scopes are queried.

Request

This request follows the QueryScope tag defined in the XSDs.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<XmlInterfaceRequest>
    <QueryScope>
      <Name>Global</Name>
    </QueryScope>
</XmlInterfaceRequest>
```

If query is successful, the list of the scopes will be returned in XML as a response. If no Ref element occurs, it means the scope is a global scope in system.

Table 13: QueryScope Operation Parameters

Parameter Name	Optional/Required	Limitation
Name	Optional	250 characters (string)

If the query is successful, the list of the roles will be returned in XML as a response.

Table 14: Output Fields for QueryScope Operation

Field Name	Description
Name	The scope's name
Description	The scope's description
ResourceControllerGroupRef	The manageable MPEs' name set.
MRAGroupRef	The manageable MRAs' name set.
NetworkElementGroupRef	The manageable Network Elements' name set.

Response

The response to this request:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConfigurationData version="7.5.0">
```

```
    <Scope>
        <Name>Global</Name>
       <Description>The default scope that automatically contains all items in the
 system</Description>
    </Scope>
</ConfigurationData>
```

A subset of the Ref element will be outputted in XML based on the mode selected by the operator. The output XML cannot be imported to the CMP again.

Chapter 9

Policy Table Interface

Topics:

Table-driven policies allows you to generalize multiple similar policies into a single policy that represents the policy structure, plus a policy table(s) that captures the differences. These tables make processes such as adding new policies, modifying existing policies, and checking consistency among policies, much simpler and less error prone.

Policy Table Overview

Polict Tables can be imported or exported to the OSSI XML Interface using the CMP. See "Importing to and Exporting from the CMP Database" in the *CMP Wireless User Guide* for details on using the OSSI XML Interface.

Policy tables can be exported to the OSSI XML Interface, and then edited or replaced. Tables can also be imported from the OSSI back to the CMP with changes, as new tables, or to delete an existing table.

The format for both export and import must be the same and must include:

- The policy table's name
- The policy table's description
- The names of the columns
- The data types of the columns
- The designation of which columns are keys
- The policy context variables for key columns
- The values in the cells

Exported Policy Table XML

The exported policy table will be in an XML file; see the example below. Descriptions of the fields are given after the table.

The XML for a policy table must maintain the integrity of the table.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<XmlInterfaceRequest>
<AddPolicyTableData>
 <PolicyTableData>
  <Name>T</Name>
  <Description>ABC</Description>
 <Table>
 <Header>
  <Field Type="smartString255" Key="true" Field="TA">TA</Field>
</Header>
 <Record Name="1330657994893">
  <Field Field="TA">TA</Field>
</Record>
 <Record Name="1330657994894">
  <Field Field="TA">TAssssss</Field>
</Record>
</Table>
</PolicyTableData>
 <PolicyTableData>
  <Name>T2</Name>
  <Description>TABLE11111111111111111111111111</Description>
 <Table>
 <Header>
  <Field Type="smartString255" Key="true" Field="TA">TA</Field>
</Header>
 <Record Name="1330657994893">
  <Field Field="TA">TA</Field>
</Record>\
```

```
<Record Name="1330657994894">
  <Field Field="TA">TAssssss</Field>
</Record>
</Table>
</PolicyTableData>
</AddPolicyTableData>
</XmlInterfaceRequest>
```

Field Name	Description
Name	The name of the Policy Table that was exported
Description	This description is copied from CMP.

Add a Policy Table

You can use the OSSI XML Interface to add a policy table, if the policy-table name does not exist in the CMP; or modify a table.

When naming a new table, the length limitation for the name is 255 characters; the description field is a free-form text that identifies the policy table and has a limit of 250 characters. If the length of either the name or description is exceeded, the import will fail.

The following example imports a Policy Table named "Sample1".

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
<AddPolicyTableData>
  <Name>Sample 1</Name>
  <Description>Sample 1</Description>
<Table>
 <Header>
  <Field Type="smartString255" Key="true" Field="TA">TA</Field>
</Header>
 <Record Name="1330657994893">
  <Field Field="TA">TA</Field>
</Record>
<Record Name="1330657994894">
  <Field Field="TA">TAssssss</Field>
</Record>
</Table>
</AddPolicyTableData>
<</XmlInterfaceRequest>
```

Modify a Policy Table

You can use the OSSI XML Interface to change a policy table that is already in CMP.

If the table name does not already exist in the CMP, the import will fail.

A column cannot be changed or removed if it is being used in a policy or policy template; if the changes in the XML change or remove such a column, the import will fail.

The import allows these types of edits:

- Editing the name and description of policy tables
- Adding, removing, and reordering columns
- Editing a column's name, data type, and key status
- Adding, removing, and reordering rows
- Changing the data within the cells

The following example imports changes for a Policy Table named "Sample 1".

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
<UpdatePolicyTableData>
  <Name>Sample 1</Name>
     <Table>
<Header>
  <Field Type="smartString255" Key="true" Field="TA">TA</Field>
</Header>
 <Record Name="1330657994893">
  <Field Field="TA">TA</Field>
</Record>
<Record Name="1330657994894">
  <Field Field="TA">TAssssss</Field>
</Record>
</Table>
     </UpdatePolicyTableData>
</XmlInterfaceRequest>
```

Delete a Policy Table

You can use the OSSI XML Interface to delete a policy table that is already in CMP.

If the Policy Table is currently used in a policy or policy template, the import will fail.

The following example deletes a Policy Table named "Sample 1".

```
<?xml version="1.0" encoding="UTF-8" ?>
<XmlInterfaceRequest>
<DeletePolicyTableData>
  <Name>Sample 1</Name>
< </DeletePolicyTableData>
</XmlInterfaceRequest>
```