ERICSSON

Application Traffic Management Configuration and Operation

SYSTEM ADMINISTRATOR GUIDE

1543-CRA 119 1170/1 Uen G

Copyright

© Ericsson AB 2009–2011. All rights reserved. No part of this document may be reproduced in any form without the written permission of the copyright owner.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design and manufacturing. Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.

Trademark List

SmartEdge	is a registered trademark of Telefonaktiebolaget LM Ericsson.
NetOp	is a trademark of Telefonaktiebolaget LM Ericsson.

Contents

1 Introduction

Note: In this document and in the CLI, the terms Deep Packet Inspection (DPI) is used interchangeably with application traffic management, and encompasses DPI and heuristics.

When the SmartEdge® router detects application traffic, it applies a DPI traffic management policy. A DPI traffic management policy classifies the traffic and maps it to one or more classes. Depending on the traffic control levels that you configure, each class is associated with a set of actions that applies to all traffic mapping for that class.

The following traffic control levels are supported:

- Per class per subscriber

  Applies a set of actions to all traffic mapping to a particular class for a specified subscriber.

- Per subscriber aggregate

  Applies a set of actions to all traffic associated with a subscriber.

- Per class per subscriber group

  Applies a set of actions to all traffic mapping to a particular class for a group of subscribers.

- Per class per SmartEdge router

  Applies a set of actions to all traffic mapping to a particular class for all subscribers configured on a SmartEdge router.

Traffic management actions are applied first to classes within the subscriber traffic, and then to all subsequent traffic controls levels configured on the node.

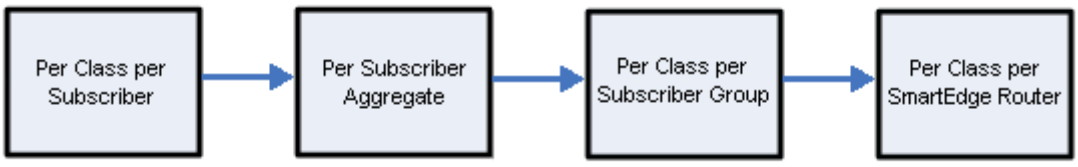

Figure 1 DPI Aggregate Traffic Control Levels

The traffic classification you use to configure per class per subscriber level traffic management is also used for all subsequent levels. In other words, if you classify "skype" traffic into class cl_01, all subsequent traffic control levels must use the same classification. It is not possible to specify different classifications for different traffic control levels.

For detailed command descriptions and usage guidelines, see Reference [6]. For overview information on the concepts presented in this document, see Reference [7].

2 Aggregating and Controlling Application Traffic According to Class and Subscriber

Configure the SmartEdge router to capture and analyze application traffic and perform per class per subscriber level control actions on the traffic by performing the following steps:

1. Configure a DPI traffic management policy.

   A DPI traffic management policy references a DPI ACL policy and DPI traffic management action policy; a DPI traffic management action policy references a DPI QoS profile.

   a Create a DPI ACL policy.

   b Create a DPI QoS profile.

   c Create a DPI traffic management action policy.

   d Create a DPI traffic management policy and a default DPI traffic management policy.

2. Assign a DPI traffic management policy to a subscriber.

The following figure illustrates the configuration workflow.

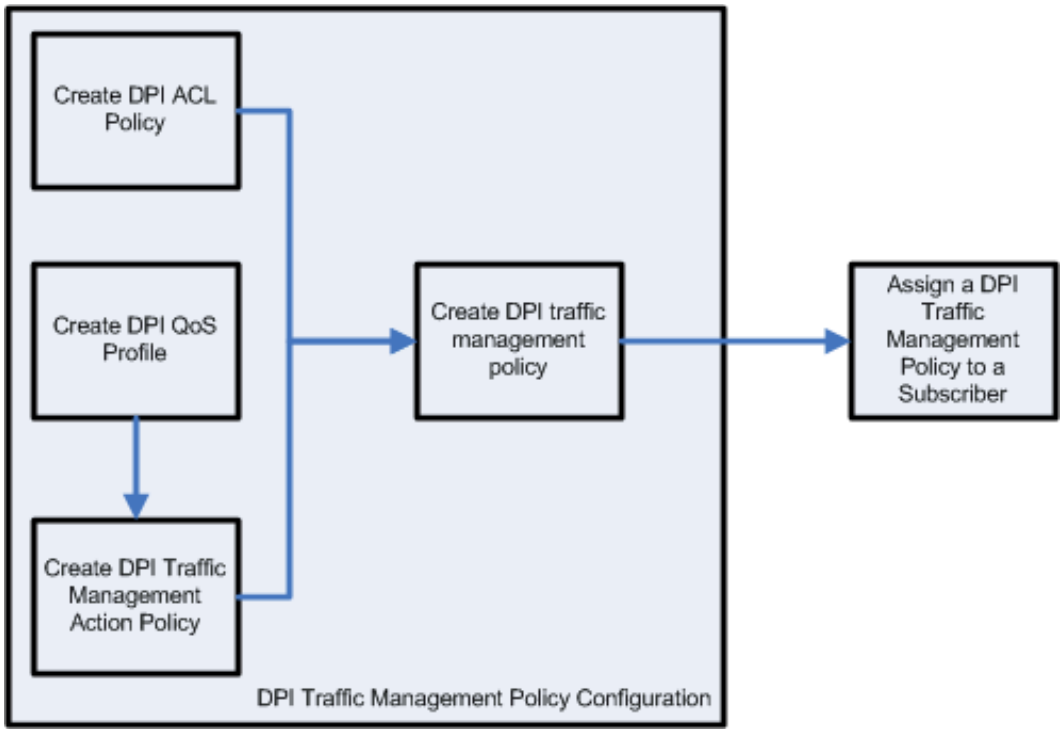

Figure 2 Per Class Per Subscriber Level Traffic Management Configuration Workflow

2.1 Configuring a DPI ACL Policy

The DPI ACL policy maps the incoming traffic to a single class value. An ACL policy uses statements to define how packets are assigned to classes. The **sequence** ***seq-num*** construct defines the sequence of the statements; if this construct is not specified, the system assigns a sequence number. A packet that does not match the criteria of the first statement is subject to the criteria of the second statement, and so on, until the end of the ACL policy is reached. The default class defined in the DPI ACL policy is used to map all traffic that was not classified into one of the other classes.

Traffic can be classified based on application protocol or transport protocol, or on application protocol category. An application or protocol category groups together applications or protocols used for a similar purpose; for example, streaming, messaging, file transfer, and so on. If a category is specified, all applications defined in the category are included.

Each application or category can be qualified with the host network, for example, BitTorrent application from a host in network 1.1.1.0/24. The IP prefix specified as the network address is matched against the destination address for inbound traffic from the subscriber and against the source address for outbound traffic to the subscriber.

2.1.1 Configuration Tasks

To configure a DPI ACL policy:

1. Create a DPI ACL policy.

   ```
   (config)#dpi access-list acl-name
   ```

2. Optional. Define a default class to which traffic is mapped if it is not classified into one of the classes defined in the DPI ACL policy.

   ```
   (dpi-acl)#default-class class-name
   ```

3. Create ACL policy statements to classify packets that meet the specified criteria.

   ```
   (dpi-acl)#[seq sequence-number] protocol protocol
   {network network-prefix/prefix-length | any} {cond
   source-port | range source-start-port source-end-port |
   any} {cond dest-port | range dest-start-port dest-end-port
   | any} class class-name
   ```

   ```
   (dpi-acl)#[seq sequence-number] protocol protocol
   {network network-prefix/prefix-length | any} [filter http
   filter-name] class class-name
   ```

   ```
   (dpi-acl)#[seq sequence-number] {application
   application-name | category category-name} [network
   network-prefix/prefix-length | any] class class-name
   ```

4. Commit the transaction.

To view configured DPI ACLs policy, enter the following command in any mode:

```
> show dpi card slot/asp-id access-list [list-name]
```

The ASE card has two ASPs, identified as 1 and 2. For information on ASE cards, ASP pools, and ASP groups, see Reference [4]. For configuration information, see Reference [2] and Reference [3].

2.1.2 Configuration Example

The following example shows how to configure the DPI ACL policy `acl_01`.

```
[local]Redback(config)#dpi access-list acl_01
[local]Redback(dpi-acl)#default-class cl_def
[local]Redback(dpi-acl)#seq 10 application bittorrent class cl_01
[local]Redback(dpi-acl)#seq 20 category streaming network 1.1.1.0/24 class cl_01
[local]Redback(dpi-acl)#seq 30 category gaming network 4.1.1.0/24 class cl_02
[local]Redback(dpi-acl)#seq 40 application skype class cl_03
```

2.2 Configuring a DPI HTTP Filter

HTTP filters are referenced in DPI ACL policies to classify HTTP traffic based on URLs, content type, HTTP operation, and extended headers. Detection is performed by meeting one or more conditions in the HTTP filter. For example, a simple HTTP filter can contain a single condition that detects a URL starting with a specified text string. An HTTP filter with multiple conditions can include URL search criteria, but also require a specific HTTP method to be detected.

When multiple conditions exist in an HTTP filter, the operation between two similar header fields is treated as a logical **or**. For example, if more than one URL condition sequence exists, the filter is satisfied if the traffic matches any of the URL conditions. Between different header fields, the operation is a logical **and**, requiring both conditions be met.

The following HTTP filter contains multiple conditions:

```
seq 10 header url is http://www.sportpunter.com
seq 20 header url contains betting
seq 30 method is get
seq 40 method is post
```

This filter is considered satisfied if it detects traffic containing either the exact URL **`http://www.sportpunter.com`** or any URL containing the string **`betting`**, and the associated HTTP method used is either **`get`** or **`post`**.

You can include up to 50 HTTP filters in an ACL and each HTTP filter can contain up to 40 conditions. If a traffic packet meets the conditions of the HTTP filter, the traffic is classified according to the class specified in the sequence definition of the ACL. Traffic management rules with regards to dropping, re-marking, or rate limiting can then be applied.

Any changes made to the configuration of an HTTP filter referenced by an ACL are effective immediately for new traffic flows.

2.2.1 HTTP Filter Configuration Tasks

To configure an HTTP filter:

1. Create an HTTP filter.

   (config)#**dpi filter http** ***filter-name***

2. Add a URL condition statement:

   seq ***sequence-number*** **header url {condition}** ***url-value*** **[case-sensitive]**

   For information on creating a URL condition statement, see Section 2.2.1.1 on page 7.

3. Add HTTP header attribute condition statements:

   seq ***sequence-number*** **header content-type {condition}** ***condition-value*** **[case-sensitive]**

   seq ***sequence-number*** **header label** ***operator-tag*** **{condition}** ***value*** **[case-sensitive]**

   seq ***sequence-number*** **header label** ***operator-tag*** **{numeric-condition}** ***numeric-value***

   seq ***sequence-number*** **method {is | is-not} {*HTTP-method*}**

   For information on creating an HTTP attribute condition statements, see Section 2.2.1.2 on page 8.

4. Commit the transaction. You can now reference the HTTP filter by name in an ACL.

To view the configured HTTP filter properties, enter the following command in any mode:

> show dpi card ***slot/asp-id*** **filter http** ***filter-name***

To view names of all configured HTTP filters:

> show dpi card ***slot/asp-id*** **filter http**

2.2.1.1 Creating a URL Condition Statement

The DPI analyzer searches packets for response and request URLs using search criteria configured in the condition statement and the detection capabilities described in Table 3.

A single HTTP filter can contain up to 40 URL condition statements.

Note: URL detection is not supported for TDM line cards.

To configure the condition statement for URLs or text-formatted HTTP header attributes, use one of the text expressions shown in Table 1.

Table 1 Conditions

Condition Type	Expressions
Text	One of the following: • **is** • **is-not** • **starts-with** • **not-starts-with** • **ends-with** • **not-ends-with** • **contains** • **not-contains**
Numeric	One of the following: • **gt**—greater than • **lt**—less than • **eq**—equal to • **neq**—not equal to

For example, a condition statement to detect Ericsson in an URL can appear as:

```
seq 20 header url contains ericsson
```

The following syntax rules apply to search criteria:

- The maximum length of the search criteria input is 255 characters.
- To detect URLs containing the **?** character, press the ESC key before the **?** when typing the search criteria.
- The **&** character is not permitted.
- A **starts-with** or **is** condition for a URL condition must begin with **http://** or **https://**
- By default the URL search criteria is case insensitive. Even when case sensitivity is explicit in a condition, the host portion of the URL remains case insensitive.

2.2.1.2 Creating an HTTP Header Attribute Condition Statement

You can create condition statements that filter HTTP traffic based on the following HTTP header attribute or methods shown in Table 2.

Table 2 HTTP Attribute and Methods Support

HTTP Attribute or Operation	Notes
Content-type	The content-type header indicates the mime type of the request.

Table 2 HTTP Attribute and Methods Support

HTTP Attribute or Operation	Notes
Extended headers	Allows detection of headers other than content-type. The following HTTP attributes cannot be configured as extended headers: • Host • Content-Type • Content-Length • Transfer-Encoding • Content-Encoding • User-Agent • URL • Method
HTTP methods	Detects traffic based on HTTP methods. Only the **`is`** and **`is not`** condition values can be used for detecting HTTP methods. Any one of the following HTTP methods can be used in a single HTTP condition: • options • get • head • post • put • delete • trace • connect

You can include multiple HTTP header attribute condition statements within the same HTTP filter, or combine HTTP header attribute conditions with URL conditions to create a more restrictive filter.

For example, the following condition statement only detects traffic with content-type of image/gif:

```
seq 20 header content-type image/gif
```

With the following configuration, either condition can be met for the HTTP filter to be satisfied:

```
seq 20 header content-type image/gif
seq 30 method is post
```

Adding a URL condition makes the filter more restrictive. The URL condition must be met and at least one of the HTTP header attribute conditions must be met:

```
seq 10 header url contains sports
seq 20 header content-type image/gif
seq 30 method is post
```

Extended headers support allows you to detect traffic containing headers other than content-type. You map extended headers using global configuration then reference an operator tag in an HTTP filter.

Note: Not all header types can be mapped as extended headers. Unsupported header types are shown in Table 2.

To map headers, use the following command in global configuration mode:

dpi traffic-management protocol http header ***header-name*** **label** ***operator-tag*** **text** | **numeric**

header-name	Name of the HTTP/1.1 header field.
operator-tag	User defined tag for the extended header
text | **numeric**	Specifies the type of extended header. Only applicable to HTTP headers not complying to RFC 2616.

To reference a mapped header in an ACL use the following sequence syntax:

[no] seq # header label ***operator-tag condition value*** **[case-sensitive]**

The ***condition*** variable corresponds to one of the condition expressions shown in Table 1.

The following example shows how to map the HTTP header Set-Cookie to a user-defined operator tag called `cookie`. Pre-defined headers have an associated data type. For headers that are not pre-defined, you must specify the type. Because Set-Cookie is a defined RFC 2616 header, you do not need to indicate the data type.

```
(config)#dpi traffic-management protocol http header
Set-Cookie label cookie
```

The following sequence in an HTTP filter detects cookies that contain the text *track-sales*:

```
[local]Redback(config)#seq 40 header label cook
ie contains track-sales
```

2.2.2 Adding an HTTP Filter Rule to an ACL

An HTTP filter rule within an ACL is a sequence with a 5-tuple rule and an associated HTTP filter. The rule maps to a class. Only one HTTP filter can

be referenced per ACL sequence. An ACL can contain a maximum of 100 sequences.

As soon as a packet is matched on the 5-tuple rule, the traffic flow is analyzed by an HTTP analyzer that collects all relevant protocol header fields. If the flow satisfies the HTTP filter conditions, the traffic is transitioned to HTTP fast path and classified under the mapped class. No further rules from the access list are analyzed. If no match is found, the next sequence is applied. If no sequence in the ACL produces a match, the traffic is classified under the default class defined at the top of the ACL.

To add an HTTP filter rule to an ACL, use the following syntax:

```
(dpi-acl)#[seq #] protocol tcp {network client-port-operator
client-port \ server-port-operator server port} {cond }
[filter http filter-name] class class-name
```

The following three types of rules exist:

- 5-tuple or network rules—sequence contains pure layer 3 and layer 4 match conditions that maps to a class.
- HTTP filter rules—sequence contains pre-analysis 5-tuple rules and an HTTP filter that together map to a class.
- Heuristic rules—sequence contains an application or categories based rule that maps to a class.

Note: All 5-tuple and HTTP filter sequences have higher priority than application/category regardless of sequence number. To avoid confusion, insert any heuristic rule sequences after 5-tuple and HTTP filter rules in an ACL as shown in Example 1.

The following example shows the configuration of an ACL named **acl1** that contains the three types of rules, including two HTTP filters:

```
dpi access-list <acl1>
  default-class cl_def
  seq 10 protocol udp network any gt 20 any  class c3
  seq 20 protocol tcp network 112.14.255.255/16 any any class c1
  seq 40 protocol tcp network 117.16.1.1/16 any eq 80 filter http u
  seq 50 protocol tcp network 119.20.1.1/16 any eq 80 filter http u
  seq 60 protocol tcp network 210.18.2.1/24 any any class c5
  seq 80 application bit-torrent class cl_h_8
  seq 100 application netbios class c_h_9
```

Example 1 ACL Configuration

Note: You cannot delete an HTTP filter that is currently referenced in an ACL.

2.2.3 Optimizing URL Detection Capabilities

URL detection applies to both HTTP and HTTPS traffic. Although HTTPS traffic is encrypted, URLs can be detected in HTTPS requests if they are sent via a proxy.

To detect URLs more effectively in both HTTPS and HTTP traffic, use the **`contains`** keyword for the condition and omit the resource prefix part of the address. Certain browsers remove the http:// or https:// part of the URL.

For example, the following condition statement detects the URL `https://www.mybank.com`:

```
seq 10 header url contains www.mybank.com
```

URL detection is dependent on the globally configured capabilities shown in Table 3. These capabilities determine the behavior of the HTTP analyzer. Using global CLI configuration, you can enable or disable certain capabilities. Other capabilities are always enabled. For more information on these global configuration commands, see the *Application Traffic Management Command Reference*.

Table 3 Supported Capabilities for DPI URL Detection

Supported Capabilities	Global CLI Configuration Syntax	Notes
HTTP pipelining	[no] dpi traffic-management protocol http pipelining	Default setting: disabled. Detects multiple requests sent in the same TCP packet or different TCP packets. Detects multiple responses sent in the same TCP packet. Enabling HTTP pipelining adversely affects performance.
HTTP header reassembly	No configuration available.	Always enabled. Detects URLs within segmented HTTP headers. URLs of 2048 bytes or less are reassembled. The task stops automatically if incomplete headers are found or the TCP connection is lost before reassembly is complete.
URL port normalization	No configuration available.	Always enabled. Performs URL normalization to determine whether the URL is equivalent to the one specified with the search criteria. For example, removes reference to port 80 if received in URLs within HTTP requests.
Escape character conversion	[no] dpi traffic-management protocol http escape-conversion	Default setting: enabled. Converts escaped characters in an HTTP URI to the corresponding UTF-8 character before comparing with URL search criteria. For example, converts "%2F" to its hexadecimal equivalent "/".

2.3 Configuring a DPI QoS Profile

QoS policies create and enforce Quality of Service (QoS) levels and bandwidth rates. A policy applies to only a particular class of packets; the class is configured using a DPI traffic management action policy, and this is referred

to as a class-based action. For more information on ACLs and QoS class definitions, marking, and rate-limiting, see Reference [8].

A DPI QoS profile handles traffic by:

- Applying rate-limiting to packets, with configurable re-mark actions for conforming and drop/remark actions for exceeding traffic
- Marking packets without rate-limiting

The above two actions are mutually exclusive. Only one marking instruction can be in effect at a time. Any succeeding marking or rate-limiting command supersedes the previous instruction.

2.3.1 Configuration Tasks

To configure a DPI QoS profile:

1. Create a DPI QoS profile.

   ```
   (config)#dpi qos profile profile-name [policing | metering]
   ```

   If you do not specify policing or metering, a bidirectional rate limiting QoS profile is implied.

2. Optional. Mark packets associated with the policy with one of the following tasks. Only one marking instruction can be in effect at any time.

   - Assign a Differentiated Services Code Point (DSCP) value.

     ```
     (dpi-qos)#mark dscp dscp-class
     ```

   - Assign a drop precedence value.

     ```
     (dpi-qos)#mark precedence prec-value
     ```

   - Assign a packet descriptor (PD) QoS priority number, a drop-precedence value, or both.

     ```
     (dpi-qos)#mark priority {group-num | ignore} [{drop-precedence {group-num | ignore} | af-drop drop-value}]
     ```

     For more information about PD QoS priority groups, see *Configuring Queuing and Scheduling*.

3. Optional. Set the policy rate for packets.

   ```
   (dpi-qos)#rate kbps {burst bytes | time-burst msec}
   ```

4. Optional. Specify the treatment of packets that conform to the set rate with one of the following tasks. Only one marking instruction can be in effect at any time.

   - Mark packets with a DSCP class.

     ```
     (dpi-qos-rate)#conform mark dscp dscp-class
     ```

   - Mark packets with a drop precedence value.

     ```
     (dpi-qos-rate)#conform mark precedence prec-value
     ```

   - Mark packets with a PD QoS priority number, a drop-precedence value, or both.

     ```
     (dpi-qos-rate)#conform mark priority {group-num | ignore} [{drop-precedence {group-num | ignore} | af-drop drop-value}]
     ```

5. Optional. Specify the treatment of packets that exceed the set rate with one of the following tasks. Only one marking instruction can be in effect at any time.

   - Drop packets.

     ```
     (dpi-qos-rate)#exceed drop
     ```

   - Mark packets with a DSCP class.

     ```
     (dpi-qos-rate)#exceed mark dscp dscp-class
     ```

   - Mark packets with a drop precedence value.

     ```
     (dpi-qos-rate)#exceed mark precedence prec-value
     ```

   - Mark packets with a PD QoS priority number, a drop-precedence value, or both.

     ```
     (dpi-qos-rate)#exceed mark priority {group-num | ignore} [{drop-precedence {group-num | ignore} | af-drop drop-value}]
     ```

6. Commit the transaction.

To view configured DPI QoS profiles, enter the following command in any mode:

```
> show dpi card slot/asp-id qos profile [profile-name]
```

2.3.2 Configuration Example

The following example shows how to configure the DPI QoS profile `qos_prof_01`.

```
[local]Redback(config)#dpi qos profile qos_prof_01
[local]Redback(dpi-qos)#rate 64 burst 3000
[local]Redback(dpi-qos-rate)#conform mark dscp df
[local]Redback(dpi-qos-rate)#exceed drop
```

The following example shows how to configure the DPI QoS profile qos_prof_02.

```
[local]Redback(config)#dpi qos profile qos_prof_02
[local]Redback(dpi-qos)#mark dscp 7
```

The following example shows how to configure the DPI QoS profile qos_prof_03.

```
[local]Redback(config)#dpi qos profile qos_prof_03 policing
[local]Redback(dpi-qos)#rate 64 burst 2000
[local]Redback(dpi-qos-rate)#exceed mark dscp 6
```

The following example shows how to configure the DPI QoS profile qos_prof_04.

```
[local]Redback(config)#dpi qos profile qos_prof_04 metering
[local]Redback(dpi-qos)#rate 64 burst 1500
[local]Redback(dpi-qos-rate)#conform mark dscp df
[local]Redback(dpi-qos-rate)#exceed mark dscp 8
```

2.4 Configuring a DPI Traffic Management Action Policy

A DPI traffic management action policy is a collection of class entries, with each class defining one or more actions for that class. Actions are applied to traffic mapped to the class through the DPI traffic management policy. Specify a class as default class to process traffic assigned to a class that is not defined in the action policy.

2.4.1 Configuration Tasks

To configure a DPI traffic management action policy:

1. Create a DPI traffic management action policy.

   ```
   (config)#dpi traffic-management action policy name
   ```

2. Optional. Specify a class as the default class to process all traffic assigned to a class that is not defined in the action policy.

   ```
   (action)#default-class class-name
   ```

3. Create a class entry to define actions to apply to traffic mapped to the class.

   ```
   (action)#class class-name
   ```

4. Optional. Apply rate-limiting or traffic impairment by applying a DPI QoS profile to traffic mapped to the class.

   ```
   (class)#qos profile profile-name [policing | metering]
   ```

   You can apply one policing and one metering QoS profile to a single class. However, you cannot apply a policing or metering QoS profile together with a bidirectional profile.

5. Optional. Generate a log when application traffic is detected in traffic mapped to the class.

   ```
   (class)#log detection
   ```

   A log is generated for every flow for every application for every subscriber, regardless of subscriber configuration. All detected flows are logged.

   Note: Logging application traffic detection should only be enabled for debugging purposes; log generation may affect system performance.

6. Optional. Drop traffic mapped to the class without rate-limiting.

   ```
   (class)#drop
   ```

7. Commit the transaction.

To view configured DPI traffic management action policies, enter the following command in any mode:

```
> show dpi card slot/asp-id traffic-management action
policy [policy-name]
```

2.4.2 Configuration Example

The following example shows how to configure the DPI traffic management action policy `acp_01`.

```
[local]Redback(config)#dpi traffic-management action policy acp_01
[local]Redback(action)#default-class cl_def
[local]Redback(action)#class cl_def
[local]Redback(class)#qos profile qos_prof_01
[local]Redback(class)#log detection
[local]Redback(class)#exit
[local]Redback(action)#class cl_01
[local]Redback(class)#qos profile qos_prof_02
[local]Redback(class)#exit
[local]Redback(action)#class cl_02
[local]Redback(class)#exit
[local]Redback(action)#class cl_03
[local]Redback(class)#qos profile qos_prof_03 policing
[local]Redback(class)#qos profile qos_prof_04 metering
```

2.5 Configuring a DPI Traffic Management Policy

A traffic management policy includes a reference to a DPI ACL policy and a traffic management action policy. The DPI ACL policy maps the traffic to a class, and each class is associated with a set of actions that applies to all traffic mapped to that class.

2.5.1 Configuration Tasks

To configure a DPI traffic management policy:

1. Configure a DPI traffic management policy.

   ```
   (config)#dpi traffic-management policy policy-name
   ```

2. Associate the DPI traffic management policy with a DPI access-list.

   ```
   (config-dpi-policy)#access-group acl-name
   ```

3. Associate the DPI traffic management policy with a DPI traffic management action policy.

   ```
   (config-dpi-policy)#action policy action-policy-name
   ```

4. Commit the transaction.

To view configured DPI traffic management policies, enter the following command in any mode:

```
> show dpi card slot/asp-id traffic-management policy
[policy-name]
```

2.5.2 Configuration Example

The following example shows how to configure the DPI traffic management policy `dpi_pol_01` and associate it with the DPI ACL policy `acl_01` and the DPI traffic management action policy `acp_01`.

```
[local]Redback(config)#dpi traffic-management policy dpi_pol_01
[local]Redback(config-dpi-policy)#access-group acl_01
[local]Redback(config-dpi-policy)#action policy acp_01
```

2.6 Configuring a Default DPI Traffic Management Policy

A global default traffic management policy is applied to traffic when the specified policy is not configured.

2.6.1 Configuration Tasks

To configure a default DPI traffic management policy:

1. Configure a global default DPI traffic management policy.

   ```
   (config)#dpi traffic-management policy default
   ```

2. Associate the DPI traffic management policy with a DPI access-list.

   ```
   (config-dpi-policy)#access-group acl-name
   ```

3. Associate the DPI traffic management policy with a DPI traffic management action policy.

   ```
   (config-dpi-policy)#action policy action-policy-name
   ```

4. Commit the transaction.

To view configured DPI traffic management policies, enter the following command in any mode:

```
> show dpi card slot/asp-id traffic-management policy
[policy-name]
```

2.6.2 Configuration Example

The following example shows how to configure the DPI traffic management policy `default` and associate it with the DPI ACL policy `acl_02` and the DPI traffic management action policy `acp_02`.

```
[local]Redback(config)#dpi traffic-management default
[local]Redback(config-dpi-policy)#access-group acl_02
[local]Redback(config-dpi-policy)#action policy acp_02
```

2.7 Assigning a DPI Traffic Management Policy to a Subscriber

The DPI traffic management policy name can be obtained through RADIUS (VSA 203 Security-Service) or configured in the subscriber record; for more information, see *RADIUS Attributes*. During the subscriber session's lifetime, the DPI traffic management policy associated with an active subscriber can be changed through RADIUS reauthentication or through Change of Authorization (CoA).

There are two ways to configure DPI traffic management for a subscriber:

- To apply a DPI policy to a subscriber, default subscriber, or subscriber profile, in the CLI, enter the **dpi traffic-management policy** ***policy-name*** command in subscriber configuration mode.
- To enable reapplying a DPI policy to a subscriber session after CoA or reauthorization, configure the Security-Service VSA in RADIUS.

To apply application traffic management to a subscriber, associate the subscriber record with a DPI traffic management policy. Different subscribers can be mapped to different DPI traffic management policies; a single traffic management policy can be used with many subscribers. Only one DPI traffic management policy can be associated with each subscriber record.

Note: NAT and DPI traffic management policies are mutually exclusive and cannot be applied together for a subscriber; NAT takes precedence over ASE security services.

2.7.1 Configuration Tasks

To apply a DPI traffic management policy through the CLI to a subscriber, default subscriber, or subscriber profile, enter the following command in subscriber configuration mode:

```
(config-sub)# dpi traffic-management policy policy-name
```

For a reauth or CoA to activate the policy, you must also configure the Security-Service VSA for this in RADIUS.

To configure the Security-Service VSA in RADIUS, perform the following steps as required:

1. To enable CoA or reauthorization for a DPI policy, configure RADIUS to send VSA 203 with the following format at initial logon for a subscriber session:

   ```
   Security-Service="dpi traffic-management enable-coa"
   ```

   This VSA must be sent at the time of initial subscriber login; else, it will not be possible to activate DPI services later on.

2. To apply a DPI policy to a subscriber through CoA or reauthorization, configure RADIUS to add the following lines to the subscriber record:

   Security-Service="dpi traffic-management enable-coa"

   Security-Service+="dpi traffic-management policy *policy-name*"

3. To associate a DPI policy with a subscriber without CoA or reauthorization support, configure RADIUS to add only the following line to the subscriber record:

   Security-Service="dpi traffic-management policy *policy-name*"

4. To delete a DPI policy from a subscriber through CoA or reauthorization, configure RADIUS to send the following line:

   Security-Service="dpi traffic-management policy"

   Either an invalid policy name or no DPI policy name sent in this line causes the policy to be deleted from the subscriber record after reauthorization.

When a DPI traffic management policy change is applied, changes to the QoS profile take effect immediately on existing flows. Other changes to the contents of the DPI traffic management action policy or the DPI traffic management ACL take effect immediately for new flows.

Note: The subscriber's context must be enabled for advanced security services. See Reference [3] for information on the **asp-group** command.

2.7.2 Configuration Example

The following example shows how to apply the DPI traffic management policy `dpi_pol_01` to subscriber `joe`.

```
[isp1]Redback(config-ctx)#subscriber name joe
[isp1]Redback(config-sub)#dpi traffic-management policy dpi_pol_01
```

2.8 Per Class Per Subscriber Level Traffic Management Example Configuration

The following example shows a full configuration of per class per subscriber level application traffic management, including a DPI ACL policy (`acl_01`), DPI QoS profiles (`qos_prof_01`, `qos_prof_02`, `qos_prof_03`), a DPI traffic management action policy (`acp_01`), and a DPI traffic management policy (`dpi_pol_01`). The policy is assigned to subscriber `joe`.

```
[local]Redback(config)#dpi access-list acl_01
[local]Redback(dpi-acl)#default-class cl_def
[local]Redback(dpi-acl)#seq 10 application bittorrent class cl_01
[local]Redback(dpi-acl)#seq 20 streaming network 1.1.1.0/24 class cl_02
[local]Redback(dpi-acl)#exit
[local]Redback(config)#dpi qos profile qos_prof_01
[local]Redback(dpi-qos)#rate 64 burst 3000
[local]Redback(dpi-qos-rate)#conform mark dscp df
[local]Redback(dpi-qos-rate)#exceed drop
[local]Redback(dpi-qos-rate)#exit
[local]Redback(dpi-qos)#exit
[local]Redback(config)#dpi qos profile qos_prof_02
[local]Redback(dpi-qos)#mark dscp 7
[local]Redback(dpi-qos)#exit
[local]Redback(config)#dpi qos profile qos_prof_03 policing
[local]Redback(dpi-qos)#rate 64 burst 2000
[local]Redback(dpi-qos-rate)#exceed mark dscp 6
[local]Redback(dpi-qos-rate)#exit
[local]Redback(dpi-qos)#exit
[local]Redback(config)#dpi qos profile qos_prof_04 metering
[local]Redback(dpi-qos)#rate 64 burst 1500
[local]Redback(dpi-qos-rate)#conform mark dscp df
[local]Redback(dpi-qos-rate)#exceed mark dscp 8
[local]Redback(dpi-qos-rate)#exit
[local]Redback(dpi-qos)#exit
[local]Redback(config)#dpi traffic-management action policy acp_01
[local]Redback(action)#default-class default
[local]Redback(action)#class cl_def
[local]Redback(class)#qos profile qos_prof_01
[local]Redback(class)#log detection
[local]Redback(class)#exit
[local]Redback(action)#class cl_01
[local]Redback(class)#qos profile qos_prof_02
[local]Redback(class)#exit
[local]Redback(action)#class cl_02
[local]Redback(class)#qos profile qos_prof_03 policing
[local]Redback(class)#qos profile qos_prof_04 metering
[local]Redback(class)#exit
[local]Redback(action)#class default
[local]Redback(action)#exit
[local]Redback(config)#dpi traffic-management policy dpi_pol_01
[local]Redback(config-dpi-policy)#access-group acl_01
[local]Redback(config-dpi-policy)#action policy acp_01
[local]Redback(config-dpi-policy)#exit
[local]Redback(config)#context isp1
[isp1]Redback(config-ctx)#subscriber name joe
[isp1]Redback(config-sub)#dpi traffic-management policy dpi_pol_01
```

3 Aggregating and Controlling Application Traffic According to Subscriber

Configure the SmartEdge router to aggregate and perform subscriber level control actions on application traffic by performing the following steps:

1. Verify that per class per subscriber level traffic management is configured on the node.

   A functional DPI traffic management policy that includes the following components is required before you can configure subscriber level traffic management:

   - DPI ACL Policy
   - DPI QoS Profile
   - DPI Traffic Management Action Policy

   To verify the existence of a valid DPI traffic management policy, enter the following command in any mode:

   > show dpi card ***slot/asp-id*** **traffic-management action policy** [***policy-name***]

   To view a valid per class per subscriber level traffic management configuration, see Section 15 on page 55.

2. Configure a subscriber DPI QoS profile.

3. Add the subscriber DPI QoS profile to a traffic management policy.

The following figure illustrates the configuration workflow:

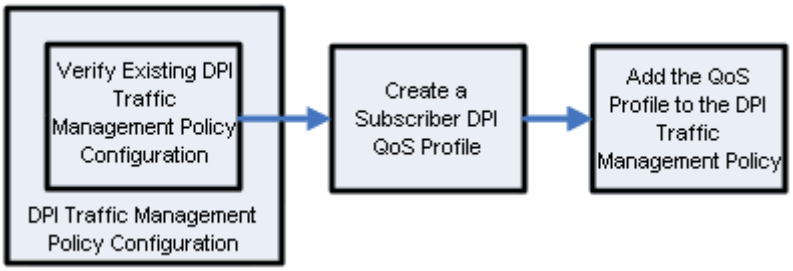

Figure 3 Subscriber Level Traffic Management Configuration Workflow

3.1 Configuring a Subscriber DPI QoS Profile

A subscriber DPI QoS profile refers to a DPI QoS profile that you apply to all traffic for a specified subscriber regardless of classification. For more information about DPI QoS profiles see Section 2.3 on page 12.

3.1.1 Configuration Tasks

To configure a subscriber DPI QoS profile, follow the same procedure used to configure a DPI QoS profile. For a detailed description of the tasks required to configure a DPI QoS profile, see Section 2.3.1 on page 13. When you configure a subscriber DPI QoS profile, remember to consider that per class per subscriber level QoS actions are applied first, followed by subscriber level QoS actions. Before you configure a subscriber QoS profile, verify that the existing per class per subscriber level QoS actions coincide with your application traffic management strategy.

To view one or all DPI QoS profiles configured on the ASE card, enter the following command in any mode:

```
> show dpi card slot/asp-id qos profile [profile-name]
```

3.1.2 Configuration Example

The following example shows how to configure the subscriber DPI QoS profile `sub_qos1`.

```
[local]Redback(config)#dpi qos profile sub_qos1
[local]Redback(dpi-qos)#rate 64 burst 3000
[local]Redback(dpi-qos-rate)#conform mark dscp df
[local]Redback(dpi-qos-rate)#exceed drop
```

3.2 Adding a Subscriber DPI QoS Profile to a DPI Traffic Management Policy

When you add a subscriber DPI QoS profile to a DPI traffic management policy, you complete the required configuration for subscriber level traffic management. You can switch between per class per subscriber level and subscriber level traffic management at any time. To enable or disable traffic management according to subscriber, add or remove the subscriber DPI QoS profile configuration from the DPI traffic management policy.

3.2.1 Configuration Tasks

To add a subscriber DPI QoS profile to a DPI traffic management policy, enter the following command in DPI traffic-management policy configuration mode.

One policing and one metering QoS profile can be applied to a single DPI traffic management policy. Neither policing nor metering QoS profiles can be applied together with a bidirectional QoS profile.

```
(config-dpi-policy)#qos profile profile-name [policing | metering]
```

Note: If the specified DPI QoS profile is not defined, the CLI is rejected.

3.2.2 Configuration Example

The following example shows how to add the subscriber DPI QoS profile `sub_qos1` to the DPI traffic management policy `p1`:

```
[local]Redback(config)#dpi traffic-management policy p1
[local]Redback(config-dpi-policy)#qos profile sub_qos1
```

The following example shows how to remove the subscriber DPI QoS profile `sub_qos1` from the DPI traffic management policy `p1`, and disable subscriber level traffic management:

```
[local]Redback(config)#dpi traffic-management policy p1
[local]Redback(config-dpi-policy)#no qos profile sub_qos1
```

3.3 Subscriber Level Traffic Management Example Configuration

The following example shows a full configuration of subscriber level traffic management, including subscriber DPI QoS profile configuration. This example implies that the DPI traffic-management policy `p1` includes a DPI ACL policy, DPI QoS profile, and a DPI traffic management action policy, and is also assigned to a subscriber. For a complete example describing how to configure per class per subscriber level traffic management, see Section 2.8 on page 20.

```
[local]Redback(config)#dpi qos profile sub_qos1
[local]Redback(dpi-qos)#rate 64 burst 3000
[local]Redback(dpi-qos-rate)#conform mark dscp df
[local]Redback(dpi-qos-rate)#exceed drop
[local]Redback(dpi-qos-rate)#commit
[local]Redback(dpi-qos-rate)#exit
[local]Redback(dpi-qos)#exit
[local]Redback(config)#dpi traffic-management policy p1
[local]Redback(config-dpi-policy)#qos profile sub_qos1
[local]Redback(config-dpi-policy)#commit
```

4 Aggregating and Controlling Application Traffic According to Class and Subscriber Group

Configure the SmartEdge router to aggregate and perform per class per subscriber group level control actions on application traffic by performing the following steps:

1. Verify that per class per subscriber level traffic management is configured on the node. The DPI ACL used to configure per class per subscriber level traffic management is also used to aggregate traffic according to class and subscriber group.

   To verify the existence of a valid per class per subscriber level configuration, enter the following command in any mode:

   > show dpi card ***slot/asp-id*** **traffic-management action policy** [***policy-name***]

   To view an example per class per subscriber level traffic management configuration, see Section 15 on page 55.

2. Configure an aggregate DPI traffic management action policy.
3. Configure an aggregate DPI traffic management policy.
4. Configure a DPI traffic management group.
5. Associate the DPI traffic management group with subscribers.

The following figure illustrates the configuration workflow:

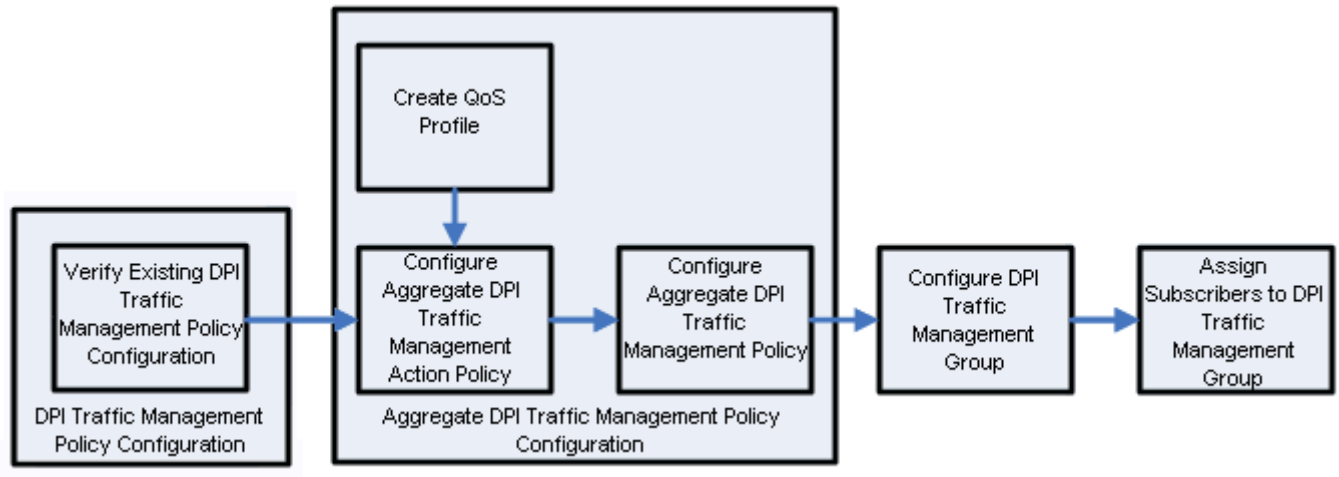

Figure 4 Per Class Per Subscriber Group Level Traffic Management Configuration Workflow

4.1 Configuring an Aggregate DPI Traffic Management Action Policy

An aggregate DPI traffic management policy is a DPI traffic management action policy that applies QoS actions to classes of traffic for groups of subscribers. Only an aggregate DPI traffic management policy can reference an aggregate DPI traffic management action policy.

4.1.1 Configuration Tasks

To configure an aggregate DPI traffic management action policy, follow the same procedure used to configure a DPI traffic management action policy. For a detailed description of the required tasks, see Section 2.5 on page 17.

Before you configure an aggregate DPI traffic management action policy, verify that the existing per class per subscriber and, if configured, any subsequent level QoS actions coincide with your application traffic management strategy.

When you configure an aggregate DPI traffic management action policy, you must use the same class values that are used to configure the DPI ACL for per class per subscriber level traffic management. If you configure an aggregate DPI traffic management action policy without using existing class values, the configuration is accepted, however, it is treated as being in error and all traffic is placed in bypass mode.

4.1.2 Configuration Example

The following example shows how to configure the aggregate DPI traffic management group `ap1agg`. In this example, the QoS profile `q10_a` is applied

to all traffic mapping to class `c10`, QoS profile `q20_a` is applied to class `c20`, and QoS profile `q30_a` is applied to all traffic mapping to class `c30`.

Note: This example implies that the QoS profiles `q10_a`, `q20_a`, and `q30_a`, are configured on the node.

```
[local]Redback(config)#dpi traffic-management action
policy ap1agg aggregate
[local]Redback(action)#class c10
[local]Redback(class)#qos profile q10_a
[local]Redback(class)#exit
[local]Redback(action)#class c20
[local]Redback(class)#qos profile q20_a
[local]Redback(class)#exit
[local]Redback(action)#class c30
[local]Redback(class)#qos profile q30_a
[local]Redback(class)#exit
[local]Redback(action)#exit
```

4.2 Configuring an Aggregate DPI Traffic Management Policy

You use an aggregate DPI traffic management policy to associate an aggregate DPI traffic management action policy with a DPI traffic management group. Because the traffic classification used for subsequent levels of traffic management is defined in the existing per class per subscriber level traffic management configuration, there is no need to provide a reference to a DPI ACL policy when you configure an aggregate DPI traffic management policy.

4.2.1 Configuration Tasks

To configure an aggregate DPI traffic management policy:

1. Create an aggregate DPI traffic management policy.

   ```
   (config)#dpi traffic-management policy policy-name
   aggregate
   ```

2. Associate an aggregate action policy with the aggregate DPI traffic management policy.

   ```
   (config-dpi-policy)#action policy policy-name aggregate
   ```

4.2.2 Configuration Example

The following example shows how to create the aggregate DPI traffic management policy `p3agg` and associate the aggregate DPI traffic management action policy `ap1agg` with it:

```
[local]Redback(config)#dpi traffic-management policy p3agg aggregate
[local]Redback(config-dpi-policy)#action policy ap1agg
```

4.3 Configuring a DPI Traffic Management Group

You use a DPI traffic management group to associate an aggregate DPI traffic management policy with a collection of subscribers.

4.3.1 Configuration Tasks

To configure a DPI traffic management group:

1. Create a DPI traffic management group.

   ```
   (config)#dpi traffic-management group group-name
   ```

   Note: Only 32 DPI traffic management groups can be created per SmartEdge router.

2. Associate a DPI traffic management policy with the group .

   ```
   (config-dpi-group)#traffic-management policy policy-name
   ```

4.3.2 Configuration Example

The following example shows how to create the DPI traffic management group `g1` and associate the DPI traffic management policy `p3agg` with it:

```
[local]Redback(config)#dpi traffic-management group g1
[local]Redback(config-dpi-group)#traffic-management policy p3agg
```

4.4 Associating a DPI Traffic Management Group with Subscribers

To finalize the per class per subscriber group level traffic management configuration, you must associate a DPI traffic management group with at least one subscriber. During the subscriber session's lifetime, you can change the DPI traffic management group associated with it through RADIUS reauthentication or CoA.

4.4.1 Configuration Tasks

To associate a DPI traffic management group with a subscriber through the CLI, enter the following command in subscriber configuration mode:

```
(config-sub)#dpi traffic-management group group-name
```

To change the group that the subscriber is associated with through RADIUS reauthorization or CoA, configure RADIUS to add the following lines to the subscriber record:

```
Security-Service="dpi traffic-management enable-coa"

Security-Service+="dpi traffic-management group group-name"
```

For more information about configuring the Security-Service VSA in RADIUS for a subscriber, see Section 2.7 on page 19.

4.4.2 Configuration Example

The following example shows how to associate subscriber `john` with traffic management group `g1`:

```
[local]Redback(config)#context local
[local]Redback(config-ctx)#subscriber name john
[local]Redback(config-sub)#dpi traffic-management group g1
```

4.5 Per Class Per Subscriber Group Level Traffic Management Example Configuration

The following example shows a full configuration of per class per subscriber group level traffic management, including QoS profile and aggregate DPI traffic management policy configuration. This example implies that the class values used to configure the DPI traffic management action policy `ap1agg` are defined in a DPI ACL used for existing per class per subscriber level traffic management. For a complete example describing how to configure per class per subscriber level traffic management, see Section 2.8 on page 20.

```
[local]Redback(config)#dpi qos profile q10_a
[local]Redback(dpi-qos)#rate 50000 burst 10000
[local]Redback(dpi-qos)#exit
[local]Redback(config)#dpi qos profile q20_a
[local]Redback(dpi-qos)#rate 60000 burst 12000
[local]Redback(dpi-qos)#exit
[local]Redback(config)#dpi qos profile q30_a
[local]Redback(dpi-qos)#rate 80000 burst 16000
[local]Redback(dpi-qos)#exit
[local]Redback(config)#dpi traffic-management action policy ap1agg aggregate
[local]Redback(action)#class c10
[local]Redback(class)#qos profile q10_a
[local]Redback(class)#exit
[local]Redback(action)#class c20
[local]Redback(class)#qos profile q20_a
[local]Redback(class)#exit
[local]Redback(action)#class c30
[local]Redback(class)#qos profile q30_a
[local]Redback(class)#exit
[local]Redback(action)#exit
[local]Redback(config)#dpi traffic-management policy p3agg aggregate
[local]Redback(config-dpi-policy)#action policy act1agg aggregate
[local]Redback(config-dpi-policy)#exit
[local]Redback(config)#dpi traffic-management group g1
[local]Redback(config-dpi-group)#traffic-management policy p3agg
[local]Redback(config-dpi-group)#exit
[local]Redback(config)#context local
[local]Redback(config-ctx)#subscriber name john
[local]Redback(config-sub)#dpi traffic-management group g1
[local]Redback(config-sub)#commit
```

5 Aggregating and Controlling Application Traffic According to Class and SmartEdge Router

Configure the SmartEdge router to aggregate and perform per class per SmartEdge router level control actions on application traffic by performing the following steps:

1. Verify that per class per subscriber level traffic management is configured on the node The DPI ACL used to configure per class per subscriber level traffic management is also used to aggregate traffic according to class and SmartEdge router.

   To verify the existence of a valid DPI traffic management policy, enter the following command in any mode:

   > show dpi card *slot/asp-id* **traffic-management action policy** [*policy-name*]

   To view an example per class per subscriber level traffic management configuration, see Section 15 on page 55.

2. Configure an aggregate DPI traffic management action policy. For more information, see Section 4.1 on page 28.

3. Configure an aggregate DPI traffic management policy. For more information, see Section 4.2 on page 29.

4. Configure the global DPI traffic management group.

The following figure illustrates the configuration workflow:

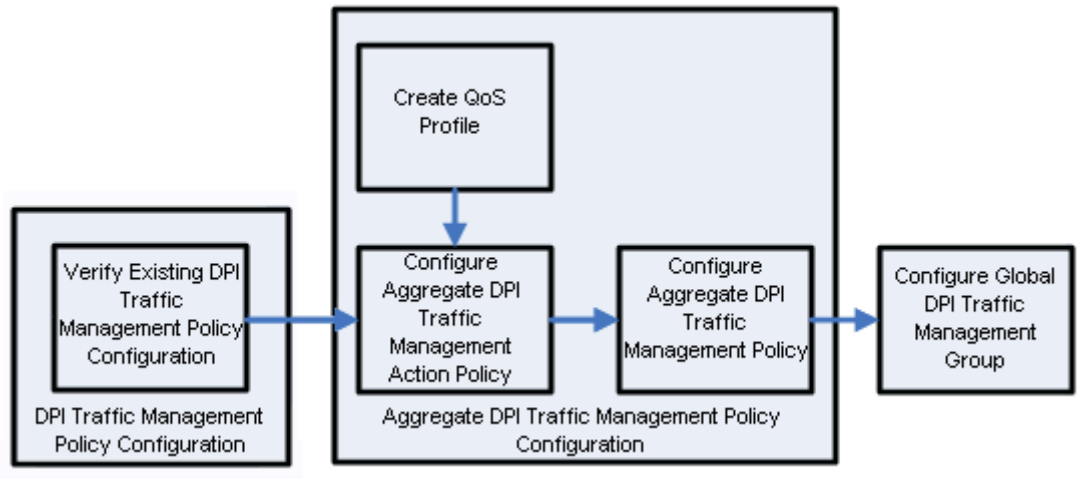

Figure 5 Per Class Per SmartEdge Router Level Traffic Management Configuration Workflow

5.1 Configuring the Global DPI Traffic Management Group

A global DPI traffic management group is similar to a DPI traffic management group, except that there is only one global group per SmartEdge router and all subscribers automatically belong to this group.

For more information about configuring a DPI traffic management group, see Section 4.3 on page 30.

5.1.1 Configuration Tasks

To configure the global DPI traffic management group, associate an aggregate DPI traffic management policy with the group by entering the following command in global DPI traffic management group configuration mode:

```
(config-dpi-group)#traffic-management policy policy-name
```

5.1.2 Configuration Example

The following example shows how to associate the DPI traffic management policy `globalagg` with the global group:

```
[local]Redback(config)#dpi traffic-management group global
[local]Redback(config)#traffic-management policy globalagg
```

5.2 Per Class Per SmartEdge Router Level Traffic Management Example Configuration

The following example shows a full configuration of per class per SmartEdge router level traffic management, including QoS profile and aggregate DPI traffic management policy configuration. This example implies that the class values used to configure the DPI traffic management action policy `pglobalagg` are defined in a DPI ACL used for existing per class per subscriber level traffic management. For a complete example describing how to configure per class per subscriber level traffic management, see Section 2.8 on page 20.

```
[local]Redback(config)#dpi qos profile q10_a
[local]Redback(dpi-qos)#rate 60000 burst 10000
[local]Redback(dpi-qos)#exit
[local]Redback(config)#dpi qos profile q20_a
[local]Redback(dpi-qos)#rate 90000 burst 12000
[local]Redback(dpi-qos)#exit
[local]Redback(config)#dpi qos profile q30_a
[local]Redback(dpi-qos)#rate 120000 burst 16000
[local]Redback(dpi-qos)#exit
[local]Redback(config)#traffic-management action policy pglobalagg
[local]Redback(action)#class c10
[local]Redback(class)#qos profile q10_se
[local]Redback(class)#exit
[local]Redback(action)#class c20
[local]Redback(class)#qos profile q20_se
[local]Redback(class)#exit
[local]Redback(action)#class c30
[local]Redback(class)#qos profile q30_se
[local]Redback(class)#exit
[local]Redback(action)#exit
[local]Redback(config)#dpi traffic-management policy globalagg aggregate
[local]Redback(config-dpi-policy)#action policy pglobalagg
[local]Redback(config-dpi-policy)#exit
[local]Redback(config)#dpi traffic-management group global
[local]Redback(config-dpi-group)#traffic-management policy globalagg
[local]Redback(config-dpi-group)#commit
```

6 Configuring Traffic Handling for Security Service Resource Failure

Certain conditions can lead to a security service resource failure; for example, an ASP run-time failure can occur if the ASE card is physically removed or develops a hardware failure.

For information on the behavior of the Advanced Services Processor (ASP) during startup, failure, and recovery, see Reference [5].

6.1 Configuration Tasks

You can configure whether the security service application drops traffic or bypasses the ASP when a resource failure occurs; by default, traffic bypasses the failed ASP.

To drop application traffic in the event of a resource failure, enter the following command in global configuration mode:

```
(config)#dpi traffic-management resource-failure-action
drop
```

7 Configuring Logging and Reporting

Reporting for advanced services like application traffic management is based on log messages. Log messages can be sent to the console, or the NetOp™ Element Management System (EMS) log mediation server and integrated with a third-party reporting solution such as Q1 Labs (http://www.q1labs.com/) or used by proprietary reporting solutions to generate deployment-specific reports. Log messages are generated when application traffic protocols are detected and to report statistics information. For information on configuring the NetOp EMS log mediation server, see Reference [9].

You can configure statistics reports to be sent to an external server at regular intervals. The ASP reports only incremental packet and byte statistics with timestamp information; all traffic-rate calculations are performed by the reporting solution.

For information on configuring logging to an external server, see Reference [3].

7.1 Enabling Statistics Collection

Statistics collection is disabled by default. To enable statistics collection, enter the following command in DPI traffic-management policy configuration mode:

```
(config-dpi-policy)#statistics [class | protocol]
```

7.2 Enabling Statistics Reporting

Statistics reporting is disabled by default. To enable per subscriber statistics reporting, enter the following command in DPI traffic-management policy configuration mode:

```
(config-dpi-policy)#accounting [class | protocol]
```

To enable per subscriber group statistics reporting, enter the following command in aggregate DPI traffic-management policy configuration mode:

```
(config-dpi-policy)#accounting class
```

Statistics are sent to the log forwarding server every 30 minutes by default.

If you enable per subscriber per protocol statistics reporting, one statistics message is sent for each application protocol detected within the configured interval. Several log messages could be sent for a subscriber at every interval.

7.3 Configuring Statistics Interval

To configure the frequency that statistics are sent to a log forwarding server, enter the following command in global configuration mode:

```
(config)#dpi traffic-management accounting interim-inter
val [minutes]
```

7.4 Configuration Example

The following example shows how to configure logging to an external server, enable per subscriber per protocol statistics collection and reporting, and also how to configure the frequency to send statistics.

```
[local]Redback(config)#asp security default
[local]Redback(config-asp-security-default)#log server 10.13.168.25 transport udp port 514
[local]Redback(config-asp-security-default)#log source 10.113.9.120
[local]Redback(config-asp-security-default)#commit
[local]Redback(config-asp-security-default)#exit
[local]Redback(config)#dpi traffic-management policy p1
[local]Redback(config-dpi-policy)#statistics protocol
[local]Redback(config-dpi-policy)#accounting protocol
[local]Redback(config)#dpi traffic-management accounting interim-interval 30
```

8 Displaying Application Traffic Management Information

Show commands display a variety of information for application traffic management. Enter show commands in any mode.

Table 4 Application Traffic Management Show Commands

To display the following information...	Enter this command...
ACLs configured on the ASE card	**show dpi card** *slot/asp-id* **access-list** [**list-name**]
One or all QoS profiles configured on the ASE card	**show dpi card** *slot/asp-id* **qos profile** [*profile-name*]
DPI traffic management action policies configured on the ASE card	**show dpi card** *slot/asp-id* **traffic-management action policy** [*policy-name*]
DPI traffic management policies configured on the ASE card	**show dpi card** *slot/asp-id* **traffic-management policy** [*policy-name*]
Global traffic management statistics	**show dpi card** *slot/asp-id* **traffic-management statistics** {**packet** [**in** | **out**] | **protocol** [*protocol-name*] | **sessions** | **signature-file** | **subscriber** | **group** *group-name* [**class** *class-name*] }
Security service specific information per subscriber	**show dpi circuit** {**agent-circuit-id** *agent-circuit-id* | **agent-remote-id** *agent-remote-id* | *slot/port*[:*chan-num*[**:***sub-chan-num*] [*circuit-id*] | **username** *subscriber*} **traffic-management** [**sessions** | **statistics sessions** | **statistics** [**packet** [**in** | **out**]] {**class** | **protocol**}]
Supported traffic management applications, categories, or signature file information on the XCRP controller card	**show dpi traffic-management** [**signature-file** *sig-filename*] [**application** | **category** [*category-name*]

To display the following information...	Enter this command...
Supported traffic management applications, categories, and their mapping on the ASP	**show dpi card** *slot/asp-id* **traffic-management** [**application** | **category** [*category-name*]]
Statistics for the ASE card, such as Rx and Tx SPI counters, system memory information, and so on	**show security card** *slot/asp-id* **statistics** {**packet** *slot* | **system**}
System-level information stored on the ASP	**show security card** *slot/asp-id* **system**
Signature file information stored on the ASP	**show dpi card** *slot/asp-id* **traffic-management signature-file**
DPI traffic management groups configured on the ASE card	**show dpi card** *slot/asp-id* **traffic-management group** [*group-name* | **global**]
Distribution of subscribers per group per ASP	**show dpi traffic-management group** [*group-name*] **distribution**
HTTP filter information stored on the ASP	**show dpi card** *slot/asp-id* **filter** [*filter-name*]
Load metrics for a DPI instance	**show dpi card** *slot/asp-id* **traffic-management statistics packet** {**instance** [**all**|**id**]}

9 Dynamically Updating the P2P Signature File

The P2P signature file is referenced during DPI protocol analysis to detect and identify known P2P application traffic. Each SmartEdge OS version contains a built-in signature file that is current as of the release date. As existing P2P applications evolve and new applications emerge, the built-in signature file becomes less effective. Keeping the signature file current between SmartEdge OS releases is therefore essential to performing comprehensive application traffic management.

A new signature file containing updated application information, categories, and RC4 encrypted signatures is created and made available every six to eight weeks. If there is no signature information update required, no file is released.

To keep the file current, perform the following steps:

1. Manually download the latest signature file to the XCRP.

2. Configure the signature file.

The configured signature file is saved to a protected memory area on the XCRP and the ASPs are notified of the signature file location. The ASPs download and validate the new file, then dynamically update their signature file definitions.

When you upgrade to a new SmartEdge OS version, a check is made to identify the current signature-file. If the signature-file packaged with the previous SmartEdge OS version is still being used, the new signature file with the new SmartEdge OS version is installed and the signature-file definition upgrade is forced on the ASPs. No configuration is required.

If it is determined that the signature file has been upgraded since the last SmartEdge OS version, verification is made to ensure the signature file in use is compatible with the new SmartEdge OS version. If the file is compatible, the signature file is not upgraded.

9.1 Downloading the Signature File

The signature file is available from an external server as a tarball which includes the signature file and associated release notes. The release notes specify SmartEdge OS compatibility and identify changes from the previous signature file.

To download the signature file tarball, you can use SFTP, FTP, or the SmartEdge OS copy command and copy the standalone signature file to the default XCRP directory.

For example:

```
copy scp://user@host/Signature-filename
```

The default directory for downloaded signature files is:

```
/flash/security/dpi/
```

9.2 Configuring the Signature File

The command to configure the signature file consists of specifying the filename and path. The configuration command validates the specified file, makes a compatibility check for the SmartEdge OS release, verifies file integrity, then saves the file to the protected `/flash` directory for automatic download to the ASPs.

Note: If you have downloaded the signature file to the default directory on the XCRP, no path specification is required.

To configure the signature file, enter the following command in global configuration mode:

```
(config)#dpi traffic-management signature-file sig-filename
```

The validated signature file is automatically downloaded by each ASP that has the service security tag. The file is saved to the local ASE directory. The applications, categories and signatures are extracted and the new signature set activated. If the activation of the signature file fails on the ASP, the ASP reboots and a critical event log entry is sent to the XCRP. An ASP with service security requires a valid signature file.

The filename format of the signature file is as follows:

App-Name-Major-Minor.sdf

Where *App-Name* is P2P, *Major* is the DPI engine major number, and *Minor* is the signature file release number.

10 Configuring Subscriber Allocation

Subscriber allocation optimizes performance and helps avoid overloading a single DPI instance. Subscribers are distributed by either round-robin or by adaptive subscriber allocation. When no traffic flows are detected for a subscriber, after a delay of 120 seconds, the subscriber is de-allocated from a DPI instance.

By default, subscribers are distributed across all the instances of a single ASP in a **round-robin** manner: each new subscriber is assigned to the instance with the least subscriber count. If the subscriber count per instance is identical, the new subscriber is assigned to the instance with the lowest instance number.

With **adaptive** subscriber allocation, subscribers are distributed based on adaptive instance load computation. The weighted average packet latency reflects current system load conditions. A new subscriber is allocated to the instance with the lightest load. Given equal loads, the new subscriber is assigned to the instance with the least subscriber count.

10.1 Configuration Tasks

To enable adaptive subscriber allocation, enter the following command in global configuration mode:

```
(config)#dpi traffic-management subscriber load-balancing
intra-asp adaptive
```

Use the no option to disable adaptive subscriber allocation and return to round-robin distribution:

```
(config)#no dpi traffic-management subscriber
load-balancing intra-asp adaptive
```

10.2 Monitoring Tasks

To display load metrics for a DPI instance, enter the following command in any mode:

```
> show dpi card slot/asp-id traffic-management statistics
packet
```

This command displays average packet latency, number of packets queued (maximum 300), and the peak number of packets queued.

To see the total number of subscribers including the current and peak values, enter the following command in any mode:

```
> show dpi card slot/asp-id traffic-management statistics
subscriber instance
```

11 Configuring Subscriber Session Limiting

The subscriber session limit refers to a single global value for the maximum number of TCP and UDP sessions allowed per subscriber. When you configure the subscriber session limit, you can spec‘‘ify whether packets associated with sessions that exceed the limit are dropped, or mapped to an action policy class. The sum of TCP and UDP sessions is limited to the configured value per subscriber. For example, if a ’’session limit of 300 is configured, then the sum of the TCP and UDP sessions for a subscriber is limited to 300.

11.1 Configuration Tasks

Subscriber session limiting is not enabled by default. To configure subscriber session limiting, enter the following command in global configuration mode:

```
(config)#dpi traffic-management maximum session
max-sessions [exceed class class-name]
```

When you enable session limiting, all packets associated with sessions that exceed the session limit are dropped by default. To map all packets associated with sessions that exceed the session limit to an action policy class, specify a class name with the **exceed class** *class-name* construct.

When you modify the session limit, changes to the class name on new and existing sessions take effect immediately. If you reduce the session limit value to below the existing session count, no new sessions are allowed until the session count drops below the new limit value. Existing sessions are not impacted.

11.2 Configuration Example

The following example shows how to configure a global subscriber session limit of 300. Packets associated with sessions that exceed this value are mapped to the action policy class `cl_06`.

```
[local]Redback(config)#dpi traffic-management maximum sessions
300 exceed class cl_06
```

12 Clearing Subscriber Sessions

To clear subscriber traffic management sessions, enter the following command in exec mode:

```
[local]Redback#clear dpi circuit {agent-circuit-id
agent-circuit-id | agent-remote-id agent-remote-id |
slot/port[:chan-num[:sub-chan-num] circuit-id | username
subscriber} traffic-management sessions
```

13 Clearing Statistics

To clear all peak counters and all packet or byte counters, enter the following command in exec mode:

```
[local]Redback#clear dpi card slot/port traffic-management
statistics
```

To clear all peak counters and all packet or byte counters for a specific subscriber, enter the following command in exec mode:

```
[local]Redback#clear dpi circuit {agent-circuit-id
agent-circuit-id | agent-remote-id agent-remote-id |
slot/port[:chan-num[:sub-chan-num] circuit-id | username subscriber}
traffic-management statistics
```

14 Enabling Debug Messages

To enable the generation of debug messages for the traffic management application, enter the following command in exec mode:

```
[local]Redback#debug dpi card slot/asp-id traffic-management message-type trace {buffer | console | external} [level level]
```

For troubleshooting information, see Reference [1].

15 Sample Configuration

For information on ASE cards, ASP pools, and ASP groups, see Reference [4]. For configuration information, see Reference [2] and Reference [3].

```
!
asp security default
log server 10.172.55.55 transport udp port 514
log source 10.192.22.24

!
!
!
asp pool p2p-pool service security
 asp 13/1
 asp 13/2
asp group p2p-group
 pool p2p-pool
 asp-count 2
!
!
dpi qos profile p2p-qos_gold
  rate 2000 burst 5000
    exceed drop
!
dpi qos profile p2p-qos_markcs0
  mark dscp 0
!
dpi qos profile p2p-qos_markcs1
  mark dscp 8
!
dpi qos profile p2p-qos_markcs2
  mark dscp 16
!
dpi qos profile p2p-qos_markcs3
  mark dscp 24
!
dpi qos profile p2p-qos_markcs4
  mark dscp 32
!
dpi qos profile p2p-qos_markdf
  mark dscp 0
!
dpi qos profile p2p-qos_markef
  mark dscp 46
!
dpi qos profile p2p-qos_platinum
  rate 5000 burst 5000
```

```
    exceed drop
!
!
dpi qos profile p2p-qos_rtlimit100
  rate 100 burst 5000
    exceed drop
!
dpi qos profile p2p-qos_silver
  rate 1000 burst 5000
    exceed drop
!
dpi access-list p2p-acl-profiles
  default-class p2p-class_default
  seq 10 application skype class p2p-class_skype
  seq 20 application bit-torrent class
  p2p-class_bittorrent
  seq 30 application edonkey class p2p-class_edonkey
  seq 40 application yahoo-messenger class p2p-class_ym
!
dpi access-list p2p-acl_monitor
  default-class p2p-class_default
  seq 10 application skype class p2p-class_skype
  seq 20 application bit-torrent class
  p2p-class_bittorrent
  seq 30 application edonkey class p2p-class_edonkey
  seq 40 application yahoo-messenger class p2p-class_ym
  seq 50 application http class p2p-class_http
  seq 60 application gnutella class p2p-class_gnutella
  seq 70 application windows-live-messenger class
  p2p-class_msn
  seq 80 application youtube class p2p-class_youtube
  seq 90 application imap class p2p-class_imap
  seq 100 application quick-time class p2p-class_qtime
  seq 110 protocol esp any class p2p-class_esp
  seq 120 protocol ahp any class p2p-class_ah
  seq 130 protocol esp any class p2p-class_esp
  seq 140 protocol tcp any eq 21 range 1 65535 class
  p2p-class_ftp
  seq 150 protocol icmp any class p2p-class_icmp
  seq 160 protocol tcp any eq 21 range 1 65535 class
  p2p-class_ftp21
  seq 170 category voip class p2p-class_voip
!
dpi access-list p2p-acl_monitor2
  seq 180 category gaming class p2p-class_gaming
  seq 190 category p2p class p2p-class_p2p
  seq 200 category file-transfer class p2p-class_ftp
  seq 210 category file-transfer 10.192.17.68/32 class
  p2p-class_ftp-cebox
!
dpi traffic-management action policy p2p-action_gold
```

```
  class p2p-class_default
    qos profile p2p-qos_gold

  class p2p-class_p2p
    qos profile p2p-qos_gold
    log detection

  class p2p-class_skype
    log detection

!
dpi traffic-management action policy p2p-action_platinum
  class p2p-class_default
    qos profile p2p-qos_platinum

  class p2p-class_p2p
    qos profile p2p-qos_platinum
    log detection

!
dpi traffic-management action policy p2p-action_silver
  class p2p-class_bittorrent
    log detection
    drop

  class p2p-class_default
    qos profile p2p-qos_silver

  class p2p-class_edonkey
    log detection
    drop


  class p2p-class_p2p
    qos profile p2p-qos_silver
    log detection

  class p2p-class_skype
    log detection

  class p2p-class_ym
    log detection
    drop

!
dpi traffic-management action policy p2p-action_monitor
  default-class p2p-class_default
  class p2p-class_ah
    qos profile p2p-qos_markef
    log detection
```

```
class p2p-class_bittorrent
  log detection

class p2p-class_edonkey
  drop

class p2p-class_esp
  qos profile p2p-qos_markcs0
  log detection

class p2p-class_ftp
  qos profile p2p-qos_markcs1
  log detection

class p2p-class_ftp-cebox
  log detection

class p2p-class_ftp21
  qos profile p2p-qos_markcs3
  log detection

class p2p-class_gaming
  log detection

class p2p-class_gnutella
  log detection

class p2p-class_http
  qos profile p2p-qos_markef
  log detection

class p2p-class_icmp
  qos profile p2p-qos_markcs2
  log detection

class p2p-class_imap
  log detection

class p2p-class_msn
   log detection

class p2p-class_p2p
  drop

class p2p-class_qtime
  log detection

class p2p-class_skype
  log detection

class p2p-class_voip
```

```
    log detection

  class p2p-class_ym
    log detection

  class p2p-class_youtube
    log detection

!
!
dpi traffic-management statistics
!
dpi traffic-management policy p2p-pol_gold
  action policy p2p-action_gold
  access-group p2p-acl_gold
!
dpi traffic-management policy p2p-pol_monitor
  action policy p2p-action_monitor
  access-group p2p-acl_monitor
!
dpi traffic-management policy p2p-pol_platinum
  action policy p2p-action_platinum
  access-group p2p-acl_platinum
!
dpi traffic-management policy p2p-pol_silver
  action policy p2p-action_silver
  access-group p2p-acl_silver
context local
!
!
context p2p
!
 no ip domain-lookup
!
 interface subscriber multibind
  ip address 40.1.1.1/24
  ip pool 40.1.1.0/24 name pc_pool
!
 interface to_Cisco7200
  ip address 150.10.1.1/24
 logging console
!
 subscriber name joe
   password joe
   ip address pool name pc_pool
   dpi traffic-management policy p2p-pol_monitor
!
 ip route 0.0.0.0/0 150.10.1.2
 ip route 40.0.0.0/24 150.10.1.2
!
```

```
!
 asp-group p2p-group service security
!
! ** End Context **
!
!Ethernet connectivity fault management configuration
!
!
card ge3-4-port 4
!
port ethernet 4/1
 no shutdown
 bind interface to_Cisco7200 p2p
!
!
card ge-10-port 9
!
port ethernet 9/1
 no shutdown
 encapsulation pppoe
 bind authentication chap pap context p2p
!
card ase 13
!
!
no service console-break
!
service crash-dump-dram
!
no service auto-system-recovery
!
```

16 Command Hierarchy

```
config
  dpi access-list
    application
    category
    default-class
    protocol
  dpi qos profile
    mark
    mark dscp
    mark precedence
    mark priority
    rate
      conform mark dscp
      conform mark precedence
      conform mark priority
      exceed drop
      exceed mark dscp
      exceed mark precedence
      exceed mark priority
  dpi traffic-management action policy
    class
      drop
      log detection
      qos profile
    default class
  dpi traffic-management group
  dpi traffic-management maximum sessions
  dpi traffic-management policy
    access-group
    accounting
    action policy
    qos profile
    statistics
  dpi traffic-management resource-failure-action
  dpi traffic-management accounting
  context
    subscriber
      dpi traffic-management policy
      dpi traffic-management group
  dpi filter http
exec
  clear dpi card
  clear dpi circuit traffic-management statistics
  clear dpi circuit traffic-management sessions
  debug dpi card traffic-management
```

```
all modes
  show dpi card access-list
  show dpi card qos profile
  show dpi card traffic-management action policy
  show dpi card traffic-management group
  show dpi card traffic-management policy
  show dpi card traffic-management statistics
  show dpi circuit
  show dpi traffic-management
  show dpi traffic-management group distribution
  show security card statistics
  show security card system
```

Glossary

ACL
Access Control List

ASP
Advanced Services Processor

CoA
Change of Authorization

DPI
Deep Packet Inspection

DSCP
Differentiated Services Code Point

QoS
Quality of Service

Reference List

[1] *ASE Troubleshooting Guide*, 12/154 51-CRA 119 1170/1

[2] *Advanced Services Configuration and Operation Using the NetOp EMS Software*, 1553-CRA 119 1170/1

[3] *Advanced Services Configuration and Operation Using the SmartEdge OS CLI*, 1/1543-CRA 119 1170/1-V1

[4] *Advanced Services Infrastructure Overview*, 1/221 02-CRA 119 1170/1-V1

[5] *Advanced Services Startup, Failure and Recovery*, 1/1553-CRA 119 1170/1-V1

[6] *Application Traffic Management Command Reference*, 190 80-CRA 119 1170/1-V1

[7] *Application Traffic Management Overview*, 221 02-CRA 119 1170/1-V1

[8] *Configuring Rate-Limiting and Class-Limiting*, 55/1543-CRA 119 1170/1-V1

[9] *Log Mediation Server*, 1/1553-CRA 119 1171/1