

# Configuring Policy Management

This chapter provides information for configuring policy-based routing management on your ERX system.

Topic	Page
Overview	1-1
References	1-7
Configuration Tasks	1-8
Creating a Rate Limit Profile	1-8
Creating Classifier Control Lists	1-17
Creating a Policy List	1-20
Creating Policy Rules	1-21
Applying a Policy List to an Interface	1-25
Enabling IP Options Filtering	1-26
Policy Applications	1-27
Monitoring Policy Management	1-33

## Overview

---

Policy management allows network service providers to implement packet forwarding and routing specifically tailored to their customers' requirements. Using policy management, you can implement policies that selectively cause packets to take different paths.

Policy management enables you to selectively forward and route packets according to your requirements. It uses policy routing to predefine packet flow to a destination port without performing a routing table lookup. Packets are sorted according to protocol or precedence into packet flows

at ingress or egress by classifier control lists (CLACLs). Policy lists initiate actions specified by rules that can include CLACLs.

Policy management provides the following types of services:

- Policy routing – Predefines a classified packet flow to a destination port or IP address. The ERX system does not perform a routing table lookup on the packet. On ingress, the packets are classified into a packet flow and sent to the preconfigured destination port. See the **next-interface** command for more details.
- QoS classification and marking – Marks packets of a packet flow. See *Creating Classifier Control Lists, later in this chapter*.
- Packet forwarding – Allows forwarding of a packet flow. See the **forward** command.
- Packet filtering – Drops packets of a packet flow. See the **filter** command.
- Packet logging – Logs packets of a packet flow. See the **log** command.
- Rate limiting – Enforces line rates below the physical line rate of the port and sets limits on packet flows. See *Creating a Rate Limit Profile, later in this chapter*.
- RADIUS policy support – Allows you to attached a preconfigured policy to an interface through RADIUS.

### *Policy Lists*

The main tool for implementing policy management is the *policy list*. A policy list is a set of *rules*, each of which initiates a *policy action*. A rule is a policy action optionally combined with a *classification*.

You can apply policy lists to packets:

- Arriving at an interface (input policy)
- Arriving at the interface but addressed to a local interface (local input policy)
- Leaving an interface (output policy)

Table 1-1 shows the commands supported for each type of policy.

**Table 1-1** Supported policy action commands

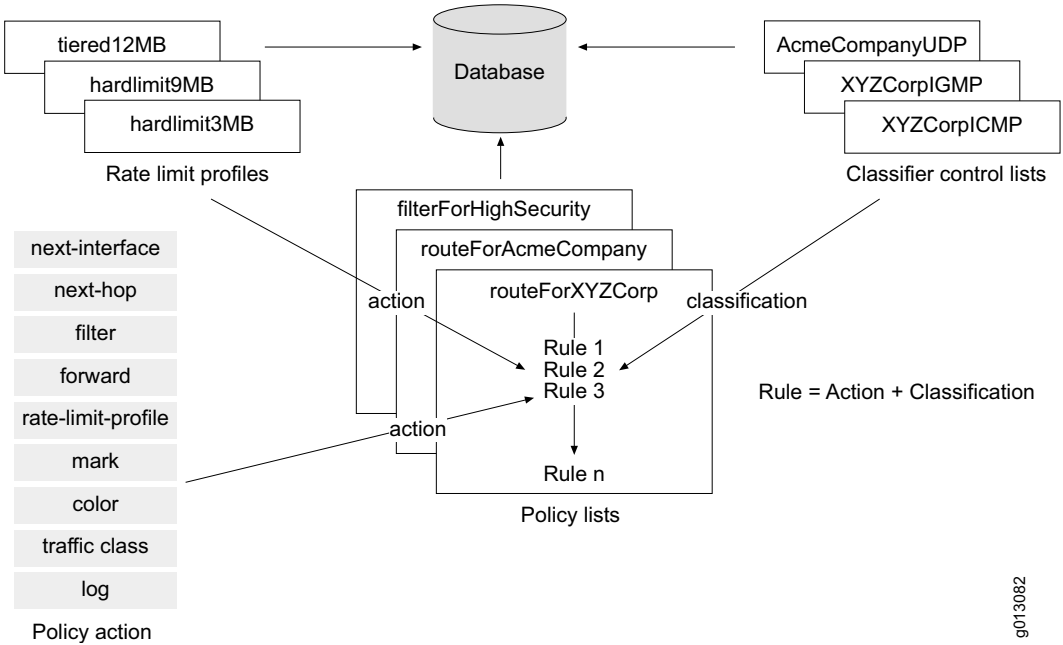
Policy Commands	Input	Local Input	Output
next-interface <sup>a</sup>	Yes	No	Yes
next-hop <sup>a</sup>	Yes	No	Yes
filter	Yes	Yes	Yes
forward	Yes	Yes	Yes
rate-limit-profile	Yes	No	Yes
mark	Yes	No	Yes
log	Yes	Yes	Yes
color	Yes	No	Yes
traffic-shape-profile	Yes	No	Yes*
traffic-class	Yes	No	Yes

a. The Fast Ethernet port on the SRP module does not support the **next-interface** or the **next-hop** policy commands.

A packet flow is specified by a CLACL that you create using the **classifier-list** command.

You create policy rules by specifying a policy action or combining policy action and a CLACL. These rules become part of a policy list that you can attach to an interface as either an input or output policy. The system implements the rules in the policy list associated with the interface.

Figure 1-1 shows how a policy list is constructed.



**Figure 1-1** Constructing a policy list

g013082

### Classifier Control Lists

CLACLs are used to specify the criteria upon which a packet flow is defined. The criteria include such packet fields as source IP address, destination IP address, source port address, destination port address, ToS byte, TCP flags, IP flags, an IP fragmentation offset.

### Rate Limit Profiles

Rate limiting is the process of limiting either a classified packet flow or source interface at a configured rate that is less than the physical rate on the port.

Use the **rate-limit-profile** command to create a rate limit profile. When you create a policy list, you can create a rule that has rate limit for an action and an association with a rate limit profile.

A rate limit profile is a set of bandwidth attributes and associated actions. Your ERX system supports the following rate-limit type(s) of rate limit profiles: one-rate and two-rate.

### One-Rate Rate Limit Profile

The one-rate rate limit profile attributes are:

- Committed rate – target rate for a packet flow
- Committed burst – amount of bandwidth allocated to accommodate bursty traffic in excess of the rate
- Excess burst – amount of bandwidth allocated to accommodate a packet in progress when the rate is in excess of the burst
- Committed action – policy action (drop, transmit, or mark) when traffic flow does not exceed the rate
- Conformed action – policy action (drop, transmit, or mark) when traffic flow exceeds the rate but not the excess burst
- Exceeded action – policy action (drop, transmit, or mark) when traffic flow exceeds the rate
- Mask value – mask of specific bits in the ToS byte when marking is the selected action

A token bucket controls how many packets per second are accepted at the configured rate. The token bucket provides flexibility in dealing with the bursty nature of data traffic. The burst sets the depth of the token bucket. The rate is the speed at which the committed token bucket is filled. The excess burst sets the extended depth of the bucket that can be used to complete an in progress packet.

At the beginning of each sample period, the bucket is filled with tokens based on the configured burst sizes. Traffic is metered to measure its volume. When traffic is received and tokens remain in the bucket, one token is removed from the bucket for every byte of data processed. As long as there are still tokens in the bucket, the traffic is treated as committed.

When the token bucket is empty, an additional number of tokens are made available to complete a packet that is in progress. After this has been depleted, traffic is treated as exceeded.

### Two-Rate Rate Limit Profile

The two-rate rate limit profile attributes are:

- Committed rate – target rate for a packet flow
- Committed burst – amount of bandwidth allocated to accommodate bursty traffic in excess of the committed rate
- Peak rate – amount of bandwidth allocated to accommodate excess traffic flow over the committed rate
- Peak burst – amount of bandwidth allocated to accommodate bursty traffic in excess of the peak rate
- Committed action – policy action (drop, transmit, or mark) when traffic flow does not exceed the committed rate
- Conformed action – policy action (drop, transmit, or mark) when traffic flow exceeds the committed rate but remains below the peak rate
- Exceeded action – policy action (drop, transmit, or mark) when traffic flow exceeds the peak rate
- Mask value – mask of specific bits in the ToS byte when marking is the selected action

The action applied by a policy to a traffic flow is determined by interaction of the rate settings and the traffic rate, as shown in Table 1-2.

**Table 1-2** Policy action applied – based on rate settings and actual traffic rate

	Committed Rate = 0	Committed Rate not 0
Peak Rate = 0	<ul style="list-style-type: none"> <li>• All traffic assigned the exceeded action.</li> </ul>	<ul style="list-style-type: none"> <li>• Traffic <math>\leq</math> committed rate assigned the committed action.</li> <li>• Traffic <math>&gt;</math> committed rate assigned the exceeded action.</li> </ul>
Peak Rate not 0	<ul style="list-style-type: none"> <li>• Traffic <math>\leq</math> peak rate assigned the conformed action.</li> <li>• Traffic <math>&gt;</math> peak rate assigned the exceeded action.</li> </ul>	<ul style="list-style-type: none"> <li>• Traffic <math>\leq</math> committed rate assigned the committed action.</li> <li>• Committed rate <math>&lt;</math> Traffic <math>&lt;</math> peak rate assigned the conformed action.</li> <li>• Traffic <math>&gt;</math> peak rate assigned the exceeded action.</li> </ul>

This implementation is known as a two-rate, three-color marking mechanism. (See RFC 2698.) Token buckets control how many packets per second are accepted at each of the configured rates. The token buckets

provide flexibility in dealing with the bursty nature of data traffic. The committed burst sets the depth of the committed token bucket. The committed rate is the speed at which the committed token bucket is filled. The peak burst sets the depth of the peak token bucket. The peak rate is the speed at which the peak token bucket are filled.

At the beginning of each sample period, the two buckets are filled with tokens based on the configured burst sizes. Traffic is metered to measure its volume. When traffic is received and tokens remain in both buckets, one token is removed from each bucket for every byte of data processed. As long as there are still tokens in the committed burst bucket, the traffic is treated as committed.

When the committed burst token bucket is empty, but tokens remain in the peak burst bucket, traffic is treated as conformed. When the peak burst token bucket is empty, traffic is treated as exceeded.

A single rate hard limit can be configured by setting the committed rate and burst rate to the desired values, the committed action to transmit, the conformed action to drop, and the exceeded action to drop. The peak rate must be set to zero.



**Note:** *The characteristic of the single rate hard limit can also be achieved by configuring a one-rate rate limit profile with the extended burst rate set to zero.*

## References

---

For more information about policy management, see the following resources:

- RFC 3198 – Terminology for Policy-Based Management (November 2001)
- RFC 2698 – A Two Rate Three Color Marker (September 1999)
- RFC 2697 – A Single Rate Three Color Marker (September 1999)
- RFC 2475 – An Architecture for Differentiated Services (December 1998)
- RFC 2474 – Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers (December 1998)

## Configuration Tasks

---

Several of the following tasks are optional. Perform the required tasks and also any optional tasks that you need for your policy management configuration:

- Create a policy list.
- Create one or more policy rules within the policy list.
- Attach a policy list to an interface.
- (Optional) Create a rate limit profile.
- (Optional) Create a CLACL.

### Creating a Rate Limit Profile

---

You can create one-rate or two-rate rate limit profiles. The **one-rate rate-limit-profile** command provides a hard-limit rate limiter or a TCP-friendly rate limiter mechanism. The **two-rate rate-limit-profile** command provides a two-rate, three-color marking mechanism.

#### *One-Rate*

Use the **rate-limit-profile** command with the **one-rate** keyword to specify a one-rate rate limit profile or to modify an existing one.

The following example creates the *tcpFriendly 8 Mb* rate limit profile. This rate limit profile, when included as part of a rule in a policy list, sets a TCP friendly rate for a specified flow:

```
host1(config)#rate-limit-profile tcpFriendly8Mb one-rate
host1(config-rate-limit-profile)#committed-rate 8000000
host1(config-rate-limit-profile)#committed-burst 1500000
host1(config-rate-limit-profile)#excess-burst 3000000
host1(config-rate-limit-profile)#committed-action transmit
host1(config-rate-limit-profile)#conformed-action transmit
host1(config-rate-limit-profile)#exceeded-action drop
host1(config-rate-limit-profile)#mask-val 255
host1(config-rate-limit-profile)#exit
```

#### **committed-action**

- Use to set the committed action for a rate limit profile.
- Valid committed actions are:
  - › **drop** – drop the packet
  - › **transmit** – transmit the packet

- › **mark** – mark the packet by setting the ToS byte (ToS field of the IP packet header) to the specified 8-bit value, and transmit the packet. The mark value is masked with the default 255 unless you use the **mask-val** command to specify a different mask.
- Use the **mask-val** command to specify a mask other than the default 255 to be used with the mark value.
- Packets are colored green.
- Example
 

```
host1(config-rate-limit-profile)#committed-action transmit
```
- Use the **no** version to restore the default value, **transmit**.

### ***committed-burst***

- Use to set the committed burst in bytes for a rate limit profile.
- When you specify a nonzero value for the rate, the burst size is automatically calculated for a 100-ms burst as described above for the **committed-rate** command. If the calculated burst size is less than the default value of 8 Kbytes, the default value is used.
- Example
 

```
host1(config-rate-limit-profile)#committed-burst 150000
```
- Use the **no** version to restore the default value, 8192 bytes.

### ***committed-rate***

- Use to set the committed rate in bits per second for a rate limit profile.
- When you specify a nonzero value for the committed rate, the committed burst size is calculated based on a 100-ms burst as follows:  
 committed burst in bytes = (committed rate in bps x 100 ms) / 8 bits per byte

The CLI displays committed rate in bits per second and committed burst in bytes. For example, if the rate is 8 Mbps, the burst size is 100 ms x 8 Mbps = 800,000 bits or 100,000 bytes:

$$\text{committed burst} = (8,000,000 \text{ bps} \times 100 \text{ ms}) / 8 = 100,000 \text{ bytes}$$

For this example, displaying the rate limit profile would show the following:

```
committed-rate 8000000
committed-burst 100000
```

If the calculated burst value is less than the default burst size of 8 KB, the default burst size is used. For most configurations this value should be sufficient, making it optional for you to configure a value for the associated committed burst size.

- Example
 

```
host1(config-rate-limit-profile)#committed-rate 800000
```
- Use the **no** version to restore the default value, 0.

### ***conformed-action***

- Use to set the conformed action for a rate limit profile.
- Valid committed actions are:
  - › **drop** – drop the packet
  - › **transmit** – transmit the packet
  - › **mark** – mark the packet by setting the ToS byte (ToS field of the IP packet header) to the specified 8-bit value, and transmit the packet. The mark value is masked with the default 255 unless you use the **mask-val** command to specify a different mask.
- Use the **mask-val** command to specify a mask other than the default 255 to be used with the mark value.
- Packets are colored yellow.
- Example

```
host1(config-rate-limit-profile)#conformed-action transmit
```
- Use the **no** version to restore the default value, **transmit**.

### ***exceeded-action***

- Use to set the exceeded action for a rate limit profile.
- Valid exceeded actions are:
  - › **drop** – drop the packet
  - › **transmit** – transmit the packet
  - › **mark** – mark the packet by setting the ToS byte (ToS field of the IP packet header) to the specified 8-bit value, and transmit the packet. The mark value is masked with the default 255 unless you use the **mask-val** command to specify a different mask.
- Use the **mask-val** command to specify a mask other than the default 255 to be used with the mark value.
- Packets are colored red.
- Example

```
host1(config-rate-limit-profile)#exceeded-action drop
```
- Use the **no** version to restore the default value, **drop**.

### ***excess-burst***

- Use to set the excess burst in bytes for a rate limit profile.
- Example

```
host1(config-rate-limit-profile)#excess-burst 3000000
```
- Use the **no** version to restore the default value, 0.

**mask-val**

- Use to set the mask value.
- This command is used in conjunction with the following commands:
  - › committed-action
  - › conformed-action
  - › exceeded-action
- Use the following mask values to set the appropriate bits in the ToS field of the IP packet header:
  - › IP Precedence – 0xE0 (three most significant bits)
  - › DS Field – 0xFC (six most significant bits)
- Example
 

```
host1(config-rate-limit-profile)#mask-val 255
```
- Use the **no** version to restore the default value, 255.

**rate-limit-profile one-rate**

- Use to specify a rate limit profile name and type in Global Configuration mode.
- If no keyword is provided in the field allocated for the one-rate/two-rate keyword, then the default is a two-rate rate limit profile.
- Enters Rate Limit Profile Configuration mode, from which you can configure attributes for the rate limit profile. See Table 1-2.
- If you execute a **rate-limit-profile** command with **one-rate** keyword and then type **exit**, the ERX system creates a rate limit profile with the following default values:

Policy Attribute	Default Value
type	one-rate
committed-rate	0
committed-burst	8192
excess-burst	0
committed-action	transmit
conformed-action	transmit
exceeded-action	drop
mask	255

- Example
 

```
host1(config)#rate-limit-profile tcpFriendly10Mb one-rate
```
- Use the **no** version to remove a rate limit profile.

*Two-Rate*

Use the **rate-limit-profile** command with the **two-rate** keyword to specify a two-rate rate limit profile or to modify an existing one.

The following example creates the *hardlimit9Mb* rate limit profile. This rate limit profile, when included as part of a rule in a policy list, sets a hard limit for a specified committed rate with neither peak rate nor peak burst ability:

```
host1(config)#rate-limit-profile hardlimit9Mb two-rate
host1(config-rate-limit-profile)#committed-rate 9000000
host1(config-rate-limit-profile)#committed-burst 20000
host1(config-rate-limit-profile)#committed-action transmit
host1(config-rate-limit-profile)#conformed-action drop
host1(config-rate-limit-profile)#exceeded-action drop
host1(config-rate-limit-profile)#mask-val 255
host1(config-rate-limit-profile)#exit
```

The following example modifies the *hardlimit9Mb* rate limit profile to include an exceeded action that marks the packets that exceed the peak rate. This marking action sets the DS field in the ToS byte (the six most significant bits) to the decimal value of 7 using a mask value of 0xFC:

```
host1(config)#rate-limit-profile hardlimit9Mb two-rate
host1(config-rate-limit-profile)#exceeded-action mark
host1(config-rate-limit-profile)#mask-val 252
host1(config-rate-limit-profile)#exit
```

To set IP precedence in the ToS byte, use a mask value of 0xE0, which sets the three most significant bits.

***committed-action***

- Use to set the committed action for a rate limit profile.
- Valid committed actions are:
  - › **drop** – drop the packet
  - › **transmit** – transmit the packet
  - › **mark** – mark the packet by setting the ToS byte (ToS field of the IP packet header) to the specified 8-bit value, and transmit the packet. The mark value is masked with the default 255 unless you use the **mask-val** command to specify a different mask.

**Note:** *If a rate-limit rule and a mark rule have the same classifier, and the packet being processed is classified by that classifier, the marking by the rate-limit rule (if it is configured to do this) overwrites the value written to the ToS byte by the mark rule.*

- Use the **mask-val** command to specify a mask other than the default 255 to be used with the mark value.
- Packets are colored green.

- Example  

```
host1(config-rate-limit-profile)#committed-action transmit
```
- Use the **no** version to restore the default value, **transmit**.

### **committed-burst**

- Use to set the committed burst in bytes for a rate limit profile.
- When you specify a nonzero value for the rate, the burst size is automatically calculated for a 100-ms burst as described above for the **committed-rate** command. If the calculated burst size is less than the default value of 8192 bytes, the default value is used.
- During a software upgrade, the committed burst size in a rate limit profile is automatically set to 8192 bytes if it was less than that value before the upgrade.
- Example  

```
host1(config-rate-limit-profile)#committed-burst 20000
```
- Use the **no** version to restore the default value, 8192 bytes.

### **committed-rate**

- Use to set the committed rate in bits per second for a rate limit profile.
- When you specify a nonzero value for the committed rate, the committed burst size is calculated based on a 100-ms burst as follows:  

$$\text{committed burst in bytes} = (\text{committed rate in bps} \times 100 \text{ ms}) \div 8 \text{ bits per byte}$$

The CLI displays committed rate in bits per second and committed burst in bytes. For example, if the rate is 8 Mbps, the burst size is 100 ms x 8 Mbps = 800,000 bits or 100,000 bytes:

$$\text{committed burst} = (8,000,000 \text{ bps} \times 100 \text{ ms}) \div 8 = 100,000 \text{ bytes}$$

For this example, displaying the rate limit profile would show the following:

```
committed-rate 8000000
committed-burst 100000
```

If the calculated burst value is less than the default burst size of 8 KB, the default burst size is used. For most configurations this value should be sufficient, making it optional for you to configure a value for the associated committed burst size.

- Example  

```
host1(config-rate-limit-profile)#committed-rate 9000000
```
- Use the **no** version to restore the default value, 0.

### ***conformed-action***

- Use to set the conformed action for a rate limit profile.
- Valid conformed actions are:
  - › **drop** – drop the packet
  - › **transmit** – transmit the packet
  - › **mark** – mark the packet by setting the ToS byte (ToS field of the IP packet header) to the specified 8-bit value, and transmit the packet. The mark value is masked with the default 255 unless you use the **mask-val** command to specify a different mask.
- Use the **mask-val** command to specify a mask other than the default 255 to be used with the mark value.
- Packets are colored yellow.
- Example

```
host1(config-rate-limit-profile)#conformed-action drop
```
- Use the **no** version to restore the default value, **transmit**.

### ***exceeded-action***

- Use to set the exceeded action for a rate limit profile.
- Valid exceeded actions are:
  - › **drop** – drop the packet
  - › **transmit** – transmit the packet
  - › **mark** – mark the packet by setting the ToS byte (ToS field of the IP packet header) to the specified 8-bit value, and transmit the packet. The mark value is masked with the default 255 unless you use the **mask-val** command to specify a different mask.
- Use the **mask-val** command to specify a mask other than the default 255 to be used with the mark value.
- Packets are colored red.
- Example

```
host1(config-rate-limit-profile)#exceeded-action drop
```
- Use the **no** version to restore the default value, **drop**.

### ***mask-val***

- Use to set the mask value.
- This command is used in conjunction with the following commands:
  - › **committed-action**
  - › **conformed-action**
  - › **exceeded-action**
- Use the following mask values to set the appropriate bits in the ToS field of the IP packet header:
  - › IP Precedence – 0xE0 (three most significant bits)
  - › DS Field – 0xFC (six significant bits)

- Example

```
host1(config-rate-limit-profile)#mask-val 0XFC
```

- Use the **no** version to restore the default value, 255.

### ***peak-burst***

- Use to set the peak burst in bytes for a rate limit profile.
- When you specify a nonzero value for the peak rate, the peak burst size is automatically calculated for a 100-ms burst as described above for the **peak-rate** command. If the calculated peak burst size is less than the default value of 8192 bytes, the default value is used.
- During a software upgrade, the committed burst size in a rate limit profile is automatically set to 8192 bytes if it was less than that value before the upgrade.

- Example

```
host1(config-rate-limit-profile)#peak-burst 96256
```

- Use the **no** version to restore the default value, 8192 bytes.

### ***peak-rate***

- Use to set the peak rate in bits per second for a rate limit profile.
- When you specify a nonzero value for the peak rate, the peak burst size is calculated based on a 100-ms burst as follows:

peak burst in bytes = (peak rate in bps x 100 ms) / 8 bits per byte

The CLI displays peak rate in bits per second and peak burst in bytes. For example, if the rate is 8 Mbps, the burst size is 100 ms x 8 Mbps = 800,000 bits or 100,000 bytes:

peak burst = (8,000,000 bps x 100 ms) / 8 = 100,000 bytes

For this example, displaying the rate limit profile would show the following:

```
peak-rate 8000000
peak-burst 100000
```

If the calculated peak burst value is less than the default peak burst size of 8 KB, the default burst size is used. For most configurations this value should be sufficient, making it optional for you to configure a value for the associated peak burst size.

- During a software upgrade, the peak rate in a rate limit profile is automatically set to 0 if it was nonzero but less than the committed rate before the upgrade.

- Example

```
host1(config-rate-limit-profile)#peak-rate 0
```

- Use the **no** version to restore the default value, 0.

**rate-limit-profile two-rate**

- Use to specify a rate limit profile name and type in Global Configuration mode.
- If no keyword is provided in the field allocated for the one-rate/two-rate keyword, then the default is a two-rate rate limit profile.
- Enters Rate Limit Profile Configuration mode, from which you can configure attributes for the rate limit profile.
- If you execute a **rate-limit-profile** command and then type **exit**, the ERX system creates a rate limit profile with the following default values:

Policy Attribute	Default Value
type	two-rate
committed-rate	0
committed-burst	8192
peak-rate	0
peak-burst	8192
committed-action	transmit
conformed-action	transmit
exceeded-action	drop
mask	255

- During a software upgrade, certain values are automatically set as follows:
  - › The committed burst size is set to 8192 if it was less than that value before the upgrade.
  - › The peak burst size is set to 8192 if it was less than that value before the upgrade.
  - › The peak rate is set to 0 if it was nonzero but less than the committed rate before the upgrade.
- Example
 

```
host1(config)#rate-limit-profile hardlimit9Mb two-rate
```
- Use the **no** version to remove a rate limit profile.



**Note:** Commands that you issue in Rate Limit Profile Configuration mode do not take effect until you exit from that mode.

## Creating Classifier Control Lists

---

Use the **classifier-list** command to specify a new CLACL or to modify an existing one.

You can specify a protocol number or one of the following protocol name keywords as a filter for packets.

- **ip** – matches IP protocol attributes such as source and destination IP address and IP mask
- **icmp** – matches ICMP protocol attributes such as source and destination IP address and IP mask, ICMP type and code
- **igmp** – matches IGMP protocol attributes such as source and destination IP address and IP mask, and IGMP type
- **tcp** – matches TCP protocol attributes such as source and destination IP address and IP mask, and source and destination TCP operator and port
- **udp** – matches UDP protocol attributes such as source and destination IP address and IP mask, and source and destination UDP operator and port



**Note:** You can use the **not** keyword before a protocol name to deny traffic for the specified protocol.

### Examples

The following example uses the **ip** keyword to set up a CLACL to accept traffic from all source addresses on the subnet of XYZ Corp:

```
host1(config)#classifier-list XYZCorpPermit ip 192.168.0.0  
0.0.255.255 any
```

The following example uses the **icmp** keyword to create a classifier control list that filters all ICMP echo requests headed toward an access link for XYZ Corp under a denial of service attack:

```
host1(config)#classifier-list XYZCorpIcmpEchoReqs icmp any  
any 8 0
```

The following example uses the **igmp** keyword to create a CLACL that filters all IGMP type 1 packets:

```
host1(config)#classifier-list XYZCorpIgmpType1 igmp any any  
1
```

The following example uses the **udp** keyword to create a CLACL that matches all traffic on UDP source ports greater than 100:

```
host1(config)#classifier-list XYZCorpUdp udp any gt 100  
172.17.2.1 0.0.255.255 gt 100
```

Some of the sample CLACLs described above are used in the next section to demonstrate how to create a policy list.

### ***classifier-list***

- Use to create or modify a classifier control list.

```
host1(config)#classifier-list YourListName any any
```

- Use the *notProtocol*, *notSrcIpAddr*, and *notDestIpAddr* to cause a match when those attributes in the packet being compared have different values. For example, a packet originating from 172.28.100.52 will match if you issued the following command:

```
host1(config)#classifier-list YourListName
notSourceIpAddress tcp host 192.168.30.100 any
```

- Use the *sourceQualifier* option to specify a single TCP or UDP port or a range of ports. The *sourceQualifier* option is composed of the following suboptions:

- › *portNumber* – single port number or the beginning of a range of port numbers
- › *portOperator* – one of the following:
  - **et** – equal to
  - **lt** – less than
  - **gt** – greater than
  - **ne** – not equal to
  - **range** – range of ports
- › *toPortNumber* – end of a range of port numbers

For example, the following command matches packets with source address 198.168.30.100 and UDP source port numbers in the range 1–10:

```
host1(config)#classifier-list YourListName udp host
192.168.30.100 range 1 10 any
```

- Use classifier lists containing multiple elements when you configure classification to match any of multiple IP header field combinations. The behavior of multiple-element classifier-list classification is the logical OR of the elements in the classifier control list. For example, to match all packets that have a source IP address of 192.168.30.100 or have a destination IP address of 192.168.30.200, then configure your classifier-list as follows:

```
host1(config)#classifier-list boston5 ip host 192.168.30.100
any
host1(config)#classifier-list boston5 ip any host
192.168.30.200
```

The classifier control list *boston5* matches all packets with the source IP address equal to 192.168.30.100 or with the destination IP address equal to 192.168.30.200.

- Use the keywords **tos**, **dsfield**, and **precedence** to specify the ToS byte in the IP header:
  - › **tos** – specifies the use of the whole 8 bits of the ToS byte; range is 0–255; for example:

```
host1(config)#classifier-list tos128 ip any any tos 128
```

- › **dsfield** – specifies the use of the upper 6 bits of the ToS byte; range is 0–63; for example:

```
host1(config)#classifier-list low-drop-prec ip any any
    dsfield 10
```

- › **precedence** – specifies the use of the upper 3 bits of the ToS byte; range is 0–7; for example:

```
host1(config)#classifier-list priority ip any any precedence 1
```

- Use the *destinationQualifier* option to specify a single TCP or UDP port or range of ports, an ICMP code and optional type, or an IGMP type. The *destinationQualifier* option is composed of the following suboptions:

- › *portNumber* – single port number or the beginning of a range of port numbers (TCP and UDP only)

- › *portOperator* – one of the following (TCP and UDP only):

- **et** – equal to
- **lt** – less than
- **gt** – greater than
- **ne** – not equal to
- **range** – range of ports

- › *toPortNumber* – end of a range of port numbers (TCP and UDP only)

- › *icmpType* – ICMP message type (ICMP only)

- › *icmpCode* – ICMP message code (ICMP only)

- › *igmpType* – IGMP message type (IGMP only)

For example, the following command matches packets with source address 198.168.30.100 and ICMP type 2 and code 10.

```
host1(config)#classifier-list YourListName icmp host
    192.168.30.100 any 2 10
```

- Use the **tcp-flags** keyword and a logical equation (a quotation-enclosed string using ! for NOT, & for AND) to match one or more of the following TCP flags: **ack, fin, push, rst, syn, urgent**.

```
host1(config)#classifier-list telnetConnects tcp
    192.168.10.0 0.0.0.255 host 10.10.10.10 eq 23 tcp-flags
    "syn & !ack"
```

- Use the **ip-flags** keyword and a logical equation (a quotation-enclosed string using ! for NOT, & for AND) to match one or more of the following IP flags: **dont-fragment, more-fragments, reserved**. For example:

```
host1(config)#classifier-list dontFragment ip any any
    ip-flags "dont-fragment"
```

- For both IP flags and TCP flags, if you specify only a single flag, the logical equation does not require quotation marks.
- Use the **ip-frag-offset** keyword and the **eq** or **gt** operator to match an IP fragmentation offset equal to 0 or 1, or greater than 1.

For example, the following commands configure a policy to filter fragmentation offsets equal to 1:

```
host1(config)#classifier-list fragOffsetAttack ip any host
10.10.10.10 ip-frag-offset-value eq 1
host1(config)#policy-list dosProtect
host1(config-policy)#filter classifier-group
fragOffsetAttack
host1(config-policy)#forward
host1(config-policy)#exit
```

- Use the **no** version to remove the classifier control list.

## Creating a Policy List

---

You can create a policy list with up to 512 rules in it. Each rule is composed of a policy action and, optionally, a CLACL. Policy rules are described later in the next section.

The following example demonstrates how to create the policy list *routeForXYZCorp*. Refer to previous sections for descriptions on how to create the classifier control lists and the rate limit profile used in this example.

- 1 From Global Configuration mode, create the policy list *routeForXYZCorp*:

```
host1(config)#policy-list routeForXYZCorp
```

- 2 From Policy Configuration mode, add a rule that specifies the egress ATM interface in slot 10, port 1, subinterface 1:

```
host1(config-policy)#next-interface atm 10/1.1
classifier-group XYZCorpPermit
```

- 3 Add a rule that specifies the next hop as destination IP address 192.168.25.45:

```
host1(config-policy)#next-hop 192.168.25.45
```

- 4 Add a rule that filters all ICMP echo requests headed toward an access link:

```
host1(config-policy)#filter classifier-group
XYZCorpIcmpEchoRequests
```

- 5 Add a rule that forwards all ICMP echo requests headed toward an access link:

```
host1(config-policy)#forward classifier-group
XYZCorpIcmpEchoRequests
```

- 6 Add a rule that sets the precedence for packets ToS stamped with values of 0–5:

```
host1(config-policy)#mark tos-precedence 0 classifier-group
XYZCorpPrecedence05
```

- 7 Add a rule that sets the precedence for packets ToS stamped with values of 6–7:

```
host1(config-policy)#mark tos-precedence 7 classifier-group  
XYZCorpPrecedence67
```

- 8 Add a rule that uses a rate limit profile to set a hard limit for a specified committed rate with no burst ability:

```
host1(config-policy)#rate-limit-profile hardLimit9MB
```

- 9 Display the policy list:

```
host1(config)#show policy-list
```



**Note:** Commands that you issue in Policy Configuration mode do not take effect until you exit from that mode.

### ***policy-list***

- Use to create or modify a policy list.
- If you execute a **policy-list** command and then type **exit**, the ERX system creates a policy list with no rules, the default. When no rules are found in a policy list, the ERX system automatically inserts a default filter rule. Attaching this policy list to an interface will filter all packets on that interface.

- Example

```
host1(config)#policy-list routeForXYZCorp
```

- Use the **no** version to remove a policy list.

## Creating Policy Rules

---

A policy rule is an association between a policy action and a CLACL. The CLACL defines the packet flow on which the policy action is taken. The CLACL field is optional. If no CLACL is specified then the packet flow on which the action is taken is all packets passing through the interface.

The order in which the rules are evaluated is defined by an optional precedence field. If no precedence is specified then the default precedence is 100. The rules are evaluated from lowest to highest precedence. Rules with equal precedence values are evaluated in order of creation.

Precedence allows rules within a policy to be ordered. This only has meaning if two rules have different classifiers and those classifiers overlap. If this is the case, and a packet is received that satisfies both classifiers, then only the rule action of the rule with the lower precedence value is performed. Precedence does not allow ordering of multiple rule actions for a single classifier. This ordering is not configurable. For rules with the same classifier, only those rules sharing the lowest precedence value are executed; rules with higher precedence values are ignored.

If two rules with the same classifier and precedence field cannot exist together, such as filter and forward, then the rule configured last will be marked as eclipsed. Eclipsed rules are treated the same as suspended rules. If two rules are configured with the same classifier and different precedence fields then the rule with the higher precedence field will be marked as eclipsed.

From Policy Configuration mode, you can assign a precedence value by using the **precedence** keyword with any of the policy commands.

For example:

```
host1(config-policy)#next-hop 172.18.20.54 precedence 21
```

A policy rule can be suspended by utilizing the **suspend** version of that rule. This maintains the policy rule with its current statistics, but will no longer have any impact on any packets moving through the forwarding path. Suspending a rule is a way to stop and start a rule without losing statistics.

From Policy Configuration mode, you can suspend a rule by using the **suspend** version of that policy rule command.

For example:

```
host1(config-policy)#suspend next-hop 172.18.20.54
precedence 21
```

Policy rules may be added, removed, or suspended while the policy is attached to one or more interfaces.

### **color**

- Use to explicitly color a packet as green, yellow, or red.
- Example

```
host1(config-policy)#color green classifier-group
westfordClacl precedence 110
```

- Use the **suspend** version to suspend a color rule within a policy list.
- Use the **no** version to remove the color rule from a policy list.

### **filter**

- Use to define a rule that drops all packets conforming to the specified classifier control list.
- The **filter** command can be performed while the policy list is referenced by interfaces.

- Example

```
host1(config-policy)#filter classifier-group westfordClacl  
precedence 110
```

- Use the **suspend** version to suspend a filter rule within a policy list.
- Use the **no** version to remove a filter rule from a policy list.

### **forward**

- Use to define a rule that forwards all packets conforming to the specified classifier control list.
- The **forward** command can be performed while the policy list is referenced by interfaces.

- Example

```
host1(config-policy)#forward classifier-group westfordClacl  
precedence 110
```

- Use the **suspend** version to suspend a forward rule within a policy list.
- Use the **no** version to remove a forward rule from a policy list.

### **log**

- Use to define a rule that logs all packets conforming to the specified CLACL.

- Example

```
host1(config-policy)#log classifier-group westfordClacl  
precedence 110
```

- Use the **suspend** version to suspend a log rule within a policy list.
- Use the **no** version to remove a log rule from a policy list.

### **mark**

- Use to set the ToS byte in the IP header to a specified value.
- Use the **classifier-group** option to specify the CLACL. If you do not specify a CLACL, the ERX system selects all packets from the interface associated with this policy list for this rule.
- You must specify one of the following:
  - › A ToS byte value in the range 0–255 and a mask value in the range 1–255
  - › The keyword **tos-precedence** and a value in the range 0–7
  - › The keyword **dsfield** and a value in the range 0–63
  - › The keyword **tos** and a value in the range 0–255
- Only one mask value is allowed per policy. Multiple mark rules are allowed with various mark values, but the mask for each of these rules must be the same.

- Example

```
host1(config-policy)#mark tos-precedence 3 classifier-group  
westfordClacl precedence 110
```

- Use the **suspend** version to suspend a mark rule within a policy list.
- Use the **no** version to remove the mark rule from a policy list.

### ***next-hop***

- Use to define the IP address of the next hop for a policy list.
- The SRP Ethernet port does not support the **next-hop** policy command.
- Example

```
host1(config-policy)#next-hop 10.10.10.1 classifier-group  
westfordClacl precedence 110
```

- Use the **suspend** version to suspend a next-hop rule within a policy list.
- Use the **no** version to remove a next-hop rule from a policy list.

### ***next-interface***

- Use to define an output interface for a policy list.
- IP interfaces referenced with this command can be tracked if they move. Policies attached to an interface also move if the interface moves. However, statistics are not maintained across the move.
- The SRP module does not support the **next-interface** policy command.
- Example

```
host1(config-policy)#next-interface atm 0/0.1  
classifier-group westfordClacl precedence 110
```

- Use the **suspend** version to suspend a next-interface rule within a policy list.
- Use the **no** version to remove a next-interface rule from a policy list.

### ***rate-limit-profile***

- Use to specify a rate limit profile in a policy list from Policy Configuration mode.
- You can specify up to 124 rate limit profiles per policy list.
- Example

```
host1(config-policy)#rate-limit-profile tcpFriendly8MB  
classifier-group westfordClacl precedence 110
```

- Use the **suspend** version to suspend a rate-limit-profile rule within a policy list.
- Use the **no** version to remove a rate-limit-profile from a policy list.

### ***traffic-class***

- Use to specify a traffic class in a policy list from Policy Configuration mode.
- If you apply this rule to a packet, then that packet is placed into the specified traffic class while passing through the router.
- Example

```
host1(config-policy)#traffic-class goldClass  
classifier-group westfordClacl precedence 110
```

- Use the **suspend** version to temporarily suspend the traffic class within a policy list.
- Use the **no** version to remove a traffic class from a policy list.

## Applying a Policy List to an Interface

---

You can assign a policy list to an interface. For dynamic interfaces, you can configure an IP profile to assign the policy list to an interface. In either case, you can enable or disable the recording of statistics for bytes and packets affected by the assigned policy.

The following example assigns the policy list *routeForXYZCorp* with statistics enabled to the ingress IP interface over a serial interface:

```
host1(config)#interface atm 12/0.1
host1(config-if)#ip policy input routeForXYZCorp statistics
enabled
```

The following example creates an IP profile that applies the policy list *routeForABCCorp* to the egress of an interface:

```
host1(config)#profile bostonProfile
host1(config-profile)#ip policy output routeForABCCorp
```

When you set baseline statistics you can retrieve statistics beginning at a point in time where the baselining is enabled.

The following example baselines the statistics for the attachment of the policy list *routeForXYZCorp* with statistics enabled to the ingress of an interface:

```
host1(config)#interface atm 12/0.1
host1(config-if)#ip policy input routeForXYZCorp statistics
enabled baseline enabled
```

To show baseline counters, run the **show ip interface delta** command. (The **delta** keyword indicates that the counters are baseline counters.)

```
host1#show ip interface atm 12/0.1 delta
atm12/0.1 is up, line protocol is up
  Network Protocols: IP
    Internet address is 200.200.1.1/255.255.255.0
    Broadcast address is 255.255.255.255
    Operational MTU = 9180  Administrative MTU = 0
    Operational speed = 155520000  Administrative speed = 0
    Discontinuity Time = 1251181
    Router advertisement = disabled
    Administrative debounce-time = disabled
    Operational debounce-time = disabled
    Access routing = disabled
    Multipath mode = hashed
    In Received Packets 5, Bytes 540
    In Policed Packets 0, Bytes 0
    In Error Packets 0
```

```
In Invalid Source Address Packets 0
In Discarded Packets 0
Out Forwarded Packets 5, Bytes 540
Out Scheduler Drops Packets 0, Bytes 0
Out Policed Packets 5, Bytes 540
Out Discarded Packets 0
Policy input routeForXYZCorp
Time since last baseline 00:00:22
filter
    5 Packets  540 Bytes dropped
```

### ***ip policy***

- Use to assign a policy list to an interface or add a policy list to an IP profile that assigns it to an interface.
- Specify the **input** or **output** keyword to assign the policy list to the ingress or egress of the interface.
- Specify the **local-input** keyword to assign the policy list to data that is addressed to a local interface.

Your system supports local input policy whose principal applications are to defeat denial-of-service attacks directed at a router's local IP stack and to protect a system from being inadvertently overwhelmed by legitimate local traffic.

- You can enable or disable the recording of routing statistics for bytes and packets affected by the policy.
- If you enable statistics, you can enable or disable baselining of the statistics. The system implements the baseline by reading and storing the statistics at the time the baseline is set and then subtracting this baseline whenever baseline-relative statistics are retrieved.
- Baselining must also be enabled on the interface with the appropriate **baseline** command.
- Example

```
host1(config-if)#ip policy local-input my-policy
```

- Use the **no** version to remove the association between a policy list and an interface.

## Enabling IP Options Filtering

---

You can use the **ip filter-options all** command to filter packets with IP options on an interface. When a packet arrives on an interface, the system performs a check to see if the packet contains any IP options. If it does and if IP options filtering is enabled, that packet is dropped. IP options filtering is disabled by default.

### *ip filter-options all*

- Use to enable filtering of packets with IP options.
- Example

```
host1(config-if)#ip filter-options all
```
- The **no** version disables filtering of packets with IP options.

## Policy Applications

---

The following sections describe several practical applications you can use with policy management.

### *Policy Routing*

Policy routing allows the ERX system to classify a packet on ingress to the device and make a forwarding decision based on that classification without the need to perform the normal routing table processing. This will provide superior performance for real time applications.

Policy rules are available to allow you to make a forwarding decision that includes **next-interface** and **next-hop**.

An interface with a policy containing a next-interface rule forwards all of the packets that satisfy the classification associated with that rule to the interface specified by that next interface rule.

An interface with a policy containing a next-hop rule forwards all of the packets that satisfy the classification associated with that rule to the IP address specified by that next hop rule.

For example, you can policy route packets entering a given IP interface (atm0/0.0) so that:

- Packets from one source (1.1.1.1) are forwarded out a specified interface (atm0/0.1);
- Packets from another source (2.2.2.2) are forwarded out a different specified interface (atm2/1.1); and
- All other packets are dropped.

To do this, you need to configure the following:

```
host1#configure
host1(config)#classifier-list claclA ip host 1.1.1.1 any
host1(config)#classifier-list claclB ip host 2.2.2.2 any
host1(config)#policy-list testPolicy
host1(config-policy)#next-interface classifier-group claclA
atm0/0.1
```

```

host1(config-policy)#next-interface classifier-group claclB
atm2/1.1
host1(config-policy)#filter
host1(config-policy)#exit

host1(config)#int atm 0/0.0
host1(config-subif)#ip policy input testPolicy statistics
enabled
host1(config-subif)#exit
host1(config)#exit
host1>

```

## Security

You can configure policy management to provide a level of network security. Packet flows can be selectively forwarded or dropped. The policy rules used to support this capability are **filter** and **forward**.

An interface with a policy containing a **forward** rule allows the packet flow that satisfies the classification associated with that rule to be routed by this virtual router.

An interface with a policy containing a **filter** rule drops all of the packets of the packet flow that satisfies the classification associated with that rule.

A policy with a **filter** rule can be used to stop a denial of service attack. The classifier-list associated with this **filter** rule needs to be constructed so that it isolates the attacker's traffic into a flow. The criteria for this classifier-list can be determined by analyzing the traffic being received on an interface. The *Packet Flow Monitoring* section describes how to capture packets into a log.

For example, you can route the packets entering a given IP interface (atm0/0.0) so that:

- Packets from one source (1.1.1.1) are allowed to be routed;
- TCP packets from another source (2.2.2.2) with the IP fragmentation offset set to one are dropped;
- All other TCP packets are allowed to be routed; and
- All other packets are dropped.

To do this, you need to configure the following:

```

host1#configure
host1(config)#classifier-list claclA ip host 1.1.1.1 any
host1(config)#classifier-list claclB tcp host 2.2.2.2 any
ip-frag eq 1
host1(config)#classifier-list claclC tcp any any
host1(config)#policy-list testPolicy

```

```
host1(config-policy)#forward classifier-group claclA
host1(config-policy)#filter classifier-group claclB
host1(config-policy)#forward classifier-group claclC
host1(config-policy)#filter
host1(config-policy)#exit

host1(config)#int atm 0/0.0
host1(config-subif)#ip policy input testPolicy statistics
enabled
host1(config-subif)#exit
host1(config)#exit
host1>
```

### *Bandwidth Management*

You can configure bandwidth management to enforce ingress data rates below the physical line rate of a port. You do this by rate limiting a classified packet flow at ingress. A rate limit profile is used in conjunction with a policy rate-limit-profile rule to provide this capability. The rate limit profile is used to capture the attributes of the desired rate.

You can set an action based on one rate or two rates. These actions include drop, transmit, or mark. The default is to transmit the committed and conformed packets, and to drop the exceeded packets.

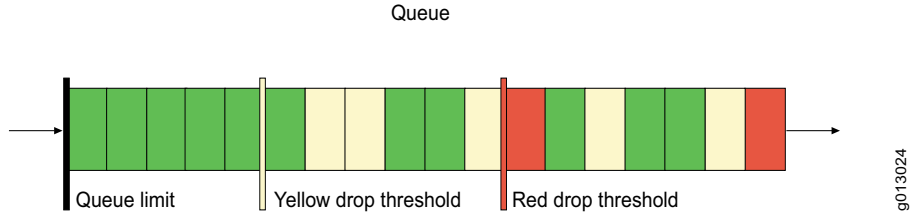
An internal color-coded tag is added automatically to each packet based on the categories:

- Committed: green
- Conformed: yellow
- Exceeded: red

The queuing system uses drop eligibility to select packets for dropping when there is congestion on an egress interface. This method is called dynamic color-based threshold dropping. Each packet queue in the system has two color-based thresholds as well as a queue limit:

- Red packets are dropped when congestion causes the queue to fill above the red threshold.
- Yellow packets are dropped when the yellow threshold is reached.
- Green packets are not dropped until the queue limit is reached.

See Figure 1-2 for an illustration of congestion management.



**Figure 1-2** Congestion management

### One-Rate Rate Limit Profile

A one-rate rate limit profile can be configured for one rate. Packets can be categorized as committed, conformed, or exceeded.

**Example 1** A one-rate rate limit profile may be configured to hard limit a packet flow to a specified rate. You can rate limit the traffic from a given source IP address (1.1.1.1) on an interface to 1M. To do this, you need to configure the following:

```

host1#configure
host1(config)#rate-limit-profile oneMegRlp one-rate
host1(config-rate-limit-profile)#committed-rate 1000000
host1(config-rate-limit-profile)#exit

host1(config)#classifier-list claclA ip host 1.1.1.1 any
host1(config)#policy-list testPolicy
host1(config-policy)#rate-limit-profile oneMegRlp
classifier-group claclA
host1(config-policy)#exit

host1(config)#interface atm 0/0.0
host1(config-subif)#ip policy input testPolicy statistics
enabled
host1(config-subif)#exit
host1(config)#exit
host1>

```

**Example 2** A one-rate rate limit profile may also be configured to provide a TCP-friendly rate limiter. To configure a rate limiter with TCP-friendly characteristics it is recommended that the committed burst be configured to allow for 1.5 seconds of data at the specified rate, and the excess burst be configured to allow 3 seconds of data at the specified committed rate.

```

host1(config)#rate-limit-profile tcpFriendly8MB one-rate
host1(config-rate-limit-profile)# committed-rate 8000000
host1(config-rate-limit-profile)# committed-burst 1500000

```

```
host1(config-rate-limit-profile)#excess-burst 3000000  
host1(config-rate-limit-profile)#committed-action transmit  
host1(config-rate-limit-profile)#exceeded-action drop  
host1(config-rate-limit-profile)#exit
```

### Two-Rate Rate Limit Profile

A two-rate rate limit profile can be configured for two different rates, committed and peak, that are used to define a two-rate, three-color marking mechanism. Packets can be categorized as committed, conformed, or exceeded:

- Up to the committed rate, packets are considered to be committed.
- From the committed to peak rate, packets are considered to be conformed.
- After the peak rate, packets are considered to be exceeded.

This configuration is implemented using token buckets. See RFC 2698 for more details.

**Example** You can rate limit the traffic from a given source IP address (1.1.1.1) on an interface such that traffic at a rate up to 1M is colored green and transmitted, traffic at a rate from 1M to 2M is colored yellow and transmitted, and traffic at a rate above 2M is dropped. To do this, you need to configure the following:

```
host1#configure  
host1(config)#rate-limit-profile oneMegRlp  
host1(config-rate-limit-profile)#committed-rate 1000000  
host1(config-rate-limit-profile)#peak-rate 2000000  
host1(config-rate-limit-profile)#committed-action transmit  
host1(config-rate-limit-profile)#conformed-action transmit  
host1(config-rate-limit-profile)#exceeded-action drop  
host1(config-rate-limit-profile)#exit  
  
host1(config)#classifier-list claclA ip host 1.1.1.1 any  
host1(config)#policy-list testPolicy  
host1(config-policy)#rate-limit-profile oneMegRlp  
          classifier-group claclA  
host1(config-policy)#exit  
  
host1(config)#interface atm 0/0.0  
host1(config-subif)#ip policy input testPolicy statistics  
          enabled
```

```
host1(config-subif)#exit
host1(config)#exit
host1>
```

## Packet Tagging

You can utilize policies to perform both in-band and out-of-band packet tagging.

In-band tagging can be done by utilizing the mark rule to modify an IP packet header ToS field. Out-of-band tagging can be done using the color or traffic class rule. The color tag affects the egress queuing threshold dropping as described in the *Bandwidth Management* section. You can use the traffic class rule to tag a packet flow so that the quality of service (QoS) application can provide traffic class queuing.

Explicit packet coloring is a service that enables you to color packets—identified by a CLACL—via a policy rule as green, yellow, or red. The ERX system uses the color to queue packets for egress queue threshold dropping. Explicit packet coloring is the same function that is provided by rate limit limiting.



**Note:** *Explicit packet coloring enables you to configure prioritized packet flows without having to configure a rate limit profile.*

### Example

If an ISP provides a B-RAS service that has both video and data components, the ISP wants to guarantee that the video traffic gets priority treatment relative to the data traffic. The ISP's users have a 1.5 Mbps VC terminating on a DSLAM. The ISP wants to allocate 800 Kbps of this link for video, if there is a video stream.

The ISP creates a classifier list to define a video packet flow, a policy to color the packets, and applies the policy to the interface:

```
host1(config)#classifier-list video ip any any dsfield 16
host1(config)#classifier-list data ip any any dsfield 32
host1(config)#policy-list colorVideoGreen
host1(config-policy)#color green classifier-group video
host1(config-policy)#color yellow classifier-group data
host1(config-policy)#exit

host1(config)#interface atm 12/1.1
host1(config-if)#ip policy input colorVideoGreen statistics
enabled
host1(config-if)#exit
host1(config)#exit
host1>#exit
```

## Packet Flow Monitoring

The policy **log** rule provides a mechanism for monitoring a packet flow. A sampling of the packets that satisfy the classification of the rule are captured in the system log.

For example, you can log ingress packets on an interface. To do this, you need to configure the following:

```
host1#configure
host1(config)#policy-list testPolicy
host1(config-policy)#log
host1(config-policy)#exit

host1(config)#int atm 0/0.0
host1(config-subif)#ip policy input testPolicy statistics
enabled
host1(config-subif)#exit

host1(config)#log destination console severity debug
host1(config)#log severity debug policyMgrPacketLog
host1(config)#log console here
host1(config)#exit
host1>
```

## Monitoring Policy Management

---

Use the following **show** commands to display statistics for a policy list associated with an interface:

- **show classifier-list**
- **show ip interface**
- **show policy-list**
- **show rate-limit-profile**

You can set a statistics baseline for policy statistics using the **ip policy** command and a **baseline interface** command. If you do not enable baselining, any **show** command output fields for baseline counters display the contents of the regular statistics counters.

You can use the output filtering feature of the **show** command to include or exclude lines of output based on a text string you specify. See *ERX System Basics Configuration Guide, Chapter 2, Command Line Interface* for details.

### *ip policy*

- Use to assign a policy list to an interface or add a policy list to an IP profile that assigns it to an interface.
- Specify the **input** or **output** keyword to assign the policy list to the ingress or egress of the interface.
- Specify the **local-input** keyword to assign the policy list to data that is addressed to a local interface.

Your ERX system supports local input policy whose principal applications are to defeat denial-of-service attacks directed at a router's local IP stack and to protect a system from being inadvertently overwhelmed by legitimate local traffic.

- You can enable or disable the recording of routing statistics for bytes and packets affected by the policy.
- If you enable statistics, you can enable or disable baselining of the statistics. The system implements the baseline by reading and storing the statistics at the time the baseline is set and then subtracting this baseline whenever baseline-relative statistics are retrieved. Unlike other baseline statistics, policy baseline statistics are not stored to nonvolatile storage (NVS).
- Baselining must also be enabled on the interface with the appropriate **baseline interface** command.
- If you issue the **baseline interface** command for an interface without first enabling policy statistics baselining on that interface, a warning message indicates "Policy baseline statistics are not enabled."
- Example

```
host1(config-if)#ip policy local-input my-policy statistics  
enabled baseline enabled
```

- Use the **no** version to remove the association between a policy list and an interface.

### *show classifier-list*

- Use to display information about CLACLs.
- Field descriptions
  - › Reference count – number of times the CLACL is referenced by policies
  - › Protocol – protocol type
  - › Source IP Address – number of the network or host from which the packet is sent
  - › Source IP Mask – mask that indicates addresses to be matched when specific bits are set
  - › Destination IP Address – number of the network or host from which the packet is sent
  - › Destination IP Mask – mask that indicates addresses to be matched when specific bits are set
- Examples

```
host1#show classifier-list athens  
athens.1 ip host 192.168.30.100 any
```

```
athens.2 ip any host 192.168.30.200
```

```
host1#show classifier-list detail
```

```
Classifier Control List Table
```

```
-----
```

```
Classifier Control List net051
```

```
Reference count:      1
```

```
Entry count:         1
```

```
Classifier-List net051.1
```

```
Protocol:             ip
Not Protocol:         false
Source IP Address:    0.0.0.0
Source IP Mask:       255.255.255.255
Not Source Ip Address: false
Destination IP Address: 10.0.51.0
Destination IP Mask:  0.0.0.255
Not Destination Ip Address: false
```

```
Classifier Control List net1151
```

```
Reference count:      1
```

```
Classifier-List net1151.1
```

```
Protocol:             ip
Not Protocol:         false
Source IP Address:    0.0.0.0
Source IP Mask:       255.255.255.255
Not Source Ip Address: false
Destination IP Address: 10.11.51.0
Destination IP Mask:  0.0.0.255
Not Destination Ip Address: false
```

### ***show ip interface***

- Use to display information about an IP interface (including policy list statistics).
- Field descriptions (related to policy management only)
  - › Network Protocols – protocols configured on the interface
  - › Internet address – IP address of the interface
  - › Broadcast address – broadcast address used by the interface
  - › Operational MTU – operational maximum transmission unit for packets sent on this interface
  - › Operational speed – speed known to the IP layer in bits per second; equal to the administrative speed if configured, otherwise inherited from the lower layer
  - › Administrative speed – configured speed known to the IP layer in bits per second

- › Discontinuity Time – time since the counters on the interface became invalid—for example, when the line card was reset
- › Router Advertisement – when enabled by the **ip irdp** command, the router advertises its presence via the ICMP Router Discovery Protocol (IRDP)
- › Administrative MTU – the administrative maximum transmission unit for packets sent on this interface
- › Operational debounce-time – the time delay that an interface must remain in a new state before the routing protocols react to the state change
- › Access routing – when enabled, an access route is installed to the host on the other end of the interface
- › In Received Packets – packets received on the interface; indicates whether packets are unicast or multicast
- › In Received Bytes – bytes received on the interface; indicates whether bytes are unicast or multicast
- › In Policed Packets – packets policed on the interface; discarded because they exceeded a traffic contract to their destination
- › In Policed Bytes – bytes policed on the interface; discarded because they exceeded a traffic contract to their destination
- › In Error Packets – packets determined to be in error at the interface
- › In Invalid Source Address Packets – packets determined to have originated from an invalid source address
- › Out Forwarded Packets – packets forwarded from the interface; indicates whether packets are unicast or multicast
- › Out Forwarded Bytes – bytes forwarded from the interface; indicates whether bytes are unicast or multicast
- › Out Scheduler Drops Packets – packets dropped by the out scheduler; indicates whether packets are committed, conformed, or exceeded
- › Out Scheduler Drops Bytes – bytes dropped by the out scheduler; indicates whether bytes are committed, conformed, or exceeded
- › Policy – indicates which policy is attached and whether it is on the input or output of the interface
- › color – explicit color applied to packet flow for queuing; green, yellow, or red
  - classifier-group – name of the classifier control list
  - Packets logged – number of packets colored
  - Bytes logged – number of bytes colored
- › bytes – number of bytes matched by the rule next-hop—address of the next-hop destination
  - classifier-group – name of the classifier control list
  - Packets transmitted – number of packets sent to the next-hop address
  - Bytes transmitted – number of bytes sent to the next-hop address
- › rate-limit-profile – name of the rate limit profile
  - classifier-group – name of the CLACL
  - committed – number of packets and bytes within the committed rate limit

- conformed – number of packets and bytes exceeding the committed rate limit but within the peak rate
  - exceeded – number of packets and bytes exceeding the peak rate
  - action – action performed on the packets matched by the rules in the rate limit profile
- Example 1

```
host1#show ip interface serial 2/1:28/24.1
serial2/1:28/24.1 is up, line protocol is up
  Network Protocols: IP
    Internet address is 172.24.1.101/255.255.255.0
    Broadcast address is 255.255.255.255
    Operational MTU = 1600 Administrative MTU = 0
    Operational speed = 155520000 Administrative speed = 0
    Discontinuity Time = 14695
    Router advertisement = disabled
    Administrative debounce-time = disabled
    Operational debounce-time = disabled
    Access routing = disabled

  In Received Packets 15, Bytes 3135
  In Policed Packets 0, Bytes 0
  In Error Packets 0
  In Invalid Source Address Packets 0
  Out Forwarded Packets 0, Bytes 0
  Out Scheduler Drops Packets 0, Bytes 0

  Policy input pl28241
  filter classifier-group clacl28241X01
    0 Packets 0 Bytes dropped
  filter classifier-group clacl28241X02
    1 Packets 202 Bytes dropped
  filter classifier-group clacl28241X03
    1 Packets 203 Bytes dropped
  filter classifier-group clacl28241X04
    1 Packets 204 Bytes dropped
  filter classifier-group clacl28241X05
    1 Packets 205 Bytes dropped
  filter classifier-group clacl28241X06
    1 Packets 206 Bytes dropped
  filter classifier-group clacl28241X07
    1 Packets 207 Bytes dropped
```

- Example 2

```
host1#show ip interface serial 2/1:2/1.101
serial2/1:2/1.101 is up, line protocol is up
  Network Protocols: IP
```

```
Internet address is 192.1.2.101/255.255.255.0  
Broadcast address is 255.255.255.255  
Operational MTU = 1600 Administrative MTU = 0  
Router advertisement = disabled  
Administrative debounce-time = disabled  
Operational debounce-time = disabled  
Access routing = disabled
```

```
In Received Packets 464, Bytes 686788  
In Policed Packets 0, Bytes 0  
In Error Packets 0  
In Invalid Source Address Packets 0  
Out Forwarded Packets 350, Bytes 256728  
Out Scheduler Drops Packets 0, Bytes 0
```

```
Policy input pl02001  
next-hop 192.2.2.201 classifier-group clacl02001  
  1 Packets 1596 Bytes transmitted  
  rate-limit-profile rlp02001 classifier-group clacl02001  
    committed: 1 Packets 1596 Bytes action: drop  
    conformed: 2 Packets 1016 Bytes action: drop  
    exceeded: 89 Packets 140956 Bytes action: drop  
next-hop 192.2.2.201 classifier-group clacl02002  
  98 Packets 144716 Bytes transmitted  
next-hop 192.2.2.201 classifier-group clacl02004  
  15 Packets 20340 Bytes transmitted  
next-hop 192.2.2.201 classifier-group clacl02005  
  20 Packets 25440 Bytes transmitted  
next-hop 192.2.2.201 classifier-group clacl02006  
  20 Packets 30440 Bytes transmitted  
  rate-limit-profile rlp02002 classifier-group clacl02002  
    committed: 98 Packets 144716 Bytes action: drop  
    conformed: 0 Packets 0 Bytes action: drop  
    exceeded: 0 Packets 0 Bytes action: drop
```

- Example 3

Assuming you have appropriately enabled policy statistics and baselining, consider the difference in standard and baselined statistics. First display standard policy statistics:

```
host1#show ip interface atm 9/1.1
```

Partial results might be as follows:

```
Policy output 2egress  
forward  
98 Packets 12544 Bytes forwarded
```

Now display baselined statistics:

```
host1#show ip interface atm 9/1.1 delta
```

Partial results might be as follows:

```
Policy output 2egress
forward
10 Packets 1280 Bytes forwarded
```

### ***show policy-list***

- Use to display information about policy lists.
- Field descriptions
  - › Policy – name of the policy list
  - › Administrative state – for SNMP use; goes to **enable** when the policy list is created. If the policy list is being modified, a user issuing the command via Telnet would see the state as disabled. Modification of a policy does not result in the modifications being applied to an interface until the administrative state is disabled and enabled.
  - › Operational status – for SNMP use; indicates whether policy has been manipulated by the ERX system to be usable by SNMP.
  - › Error Value – number of errors reported
  - › Reference count – number of attachments
  - › Referenced by interface(s) – list of interfaces to which policy is attached, indicating whether attachment is at input or output of interface
  - › Policy Rule – unique rule identifier for SNMP use
  - › Rule type – one of the following rule types:
    - Filter – rule specifies a **filter** policy action
    - Forward – rule specifies a **forward** policy action
    - Next-Interface – rule specifies a **next-interface** policy action
    - Next-Hop – rule specifies a **next-hop** policy action
    - Rate-Limit-Profile – rule specifies a **rate-limit-profile** policy action
    - Mark – rule specifies the ToS byte in the IP header to a specified value
    - Color – rule specifies the color of a packet as green, yellow, or red
    - Traffic-class – rule specifies a traffic class in a policy list
    - Log – rule specifies a **log** policy action
- Classifier control list – the name of the classifier control list attached to a rule
- Example

```
host1#show policy-list
                Policy Table
                -----
Policy westford
Administrative state: enable
Operational status:  enabled
Error Value:         0
Reference count:     2
```

```

Policy westford Rule 2 Precedence 100
  Rule type:                filter
  Rule status:              Active
  Classifier control list:  example

Referenced by interface(s):
  atm5/0.1 Input policy, Statistics enabled

Referenced by profile(s):
  leo Output policy, Statistics enabled

```

Classifier Control List Table

```

-----
example.1 ip host 192.168.30.100 any
example.2 ip any host 192.168.30.200

```

**show rate-limit-profile**

- Use to display information about rate limit profiles.
- Field descriptions
  - › Rate-Limit-Profile – name of the rate limit profile
  - › Reference Count – number of policy lists that reference this rate limit profile
  - › Committed rate – target rate for the traffic, in bits per second
  - › Committed burst – amount of bandwidth allocated to accommodate bursty traffic, in bytes
  - › Peak rate – amount of bandwidth allocated to accommodate traffic flow in excess of the committed rate, in bits per second
  - › Peak burst – amount of bandwidth allocated to accommodate bursty traffic in excess of the peak rate, in bytes
  - › Mask – value of mask applied to ToS byte in IP packet header
  - › Committed rate action – policy action (drop, transmit, or mark) taken when traffic flow does not exceed the committed rate
  - › Conformed rate action – policy action (drop, transmit, or mark) taken when traffic flow exceeds the committed rate but remains below the peak rate
  - › Exceeded rate action – policy action (drop, transmit, or mark) taken when traffic flow exceeds the peak rate
- Example

```

host1#show rate-limit-profile                                     Rate
      Limit Profile Table
      -----
Rate-Limit-Profile: rlp
  Profile Type:                one-rate
  Reference count:              0
  Committed rate:              0
  Committed burst:             8192

```

```
Excess burst: 0
Mask: 255
Committed rate action: transmit
Conformed rate action: transmit
Exceeded rate action: drop

Rate-Limit-Profile: rlp
Profile Type: two-rate
Reference count: 0
Committed rate: 0
Committed burst: 8192
Peak rate: 0
Peak burst: 8192
Mask: 255
Committed rate action: transmit
Conformed rate action: transmit
Exceeded rate action: drop
```

