Using OER to Profile the Traffic Classes

Last Updated: October 10, 2011

This module describes how Optimized Edge Routing (OER) profiles the traffic classes. To optimize traffic routing, subsets of the total traffic must be identified, and these traffic subsets are named traffic classes. The OER master controller can profile traffic classes either by manual configuration on the master controller, or by automatic learning on the basis of parameters such as throughput or delay characteristics of traffic on the border routers. Automatic learning requires traffic class parameters to be configured on the master controller.

- Finding Feature Information, page 1
- Prerequisites for Using OER to Profile the Traffic Classes, page 1
- Restrictions for Using OER to Profile the Traffic Classes, page 2
- Information About Using OER to Profile the Traffic Classes, page 2
- How to Configure OER to Profile the Traffic Classes, page 7
- Configuration Examples for Using OER to Profile the Traffic Classes, page 37
- Where To Go Next, page 40
- Additional References, page 40
- Feature Information for Using OER to Profile the Traffic Classes, page 41

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Using OER to Profile the Traffic Classes

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

- Before implementing the OER profile phase, you need to understand an overview of how OER works and how to set up OER network components. See the Cisco IOS Optimized Edge Routing Overview and Setting Up OER Network Components modules for more details.
- Cisco Express Forwarding (CEF) must be enabled on all participating devices. No other switching path is supported, even if otherwise supported by PBR.

Restrictions for Using OER to Profile the Traffic Classes

If any of the border routers is a Cisco Catalyst 6500 switch or a Cisco 7600 series router, there are some hardware constraints and the master controller will set the monitoring mode to special where only the throughput method of learning is used to profile the traffic classes. If both delay and throughput are configured, the master controller will ignore the delay configuration. For more details about the special monitoring mode, see the Measuring the Traffic Class Performance and Link Utilization Using OER module for more details.

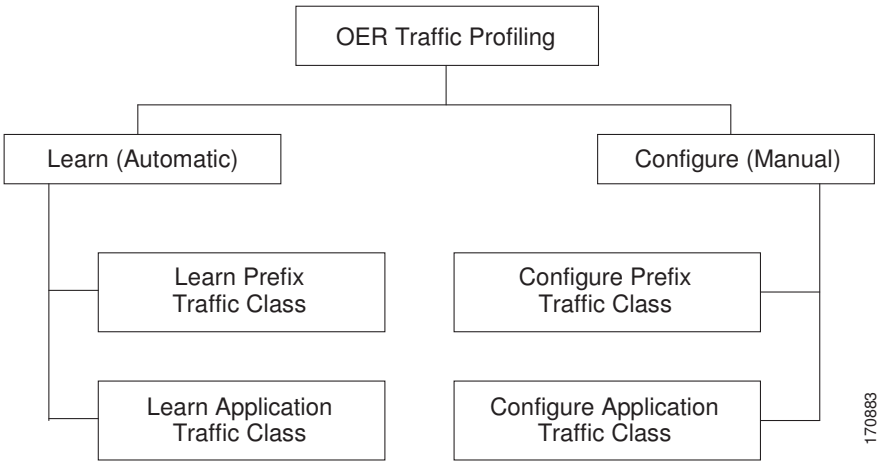
Information About Using OER to Profile the Traffic Classes

- OER Traffic Class Profiling, page 2
- OER Automatic Traffic Class Learning, page 3
- OER Manual Traffic Class Configuration, page 5

OER Traffic Class Profiling

Before optimizing traffic, OER has to determine the traffic classes from the traffic flowing through the border routers. To optimize traffic routing, subsets of the total traffic must be identified, and these traffic subsets are named traffic classes. The list of traffic classes entries is named a Monitored Traffic Class (MTC) list. The entries in the MTC list can be profiled either by automatically learning the traffic flowing through the device or by manually configuring the traffic classes. Learned and configured traffic classes can both exist in the MTC list at the same time. The OER profile phase includes both the learn mechanism and the configure mechanism. The overall structure of the OER traffic class profile process and its component parts can be seen in the the diagram below.

Figure 1 OER Traffic Class Profiling Process



The ultimate objective of this phase is to select a subset of traffic flowing through the network. This subset of traffic--the traffic classes in the MTC list--represents the classes of traffic that need to be routed based on the best performance path available.

OER Automatic Traffic Class Learning

OER can automatically learn the traffic classes while monitoring the traffic flow through border routers. Although the goal is to optimize a subset of the traffic, you may not know all the exact parameters of this traffic and OER provides a method to automatically learn the traffic and create traffic classes by populating the MTC list. Several features have been added to OER since the original release to add functionality to the automatic traffic class learning process.

Within the automatic traffic class learning process there are now three components. One component describes the automatic learning of prefix-based traffic classes, the second component describes automatic learning of application-based traffic classes, and the third component describes the use of learn lists to categorize both prefix-based and application-based traffic classes. These three components are described in the following sections:

- Prefix Traffic Class Learning Using OER, page 3
- Application Traffic Class Learning Using OER, page 4
- Learn List Configuration Mode, page 5

Prefix Traffic Class Learning Using OER

The OER master controller can be configured, using NetFlow Top Talker functionality, to automatically learn prefixes based on the highest outbound throughput or the highest delay time. Throughput learning measures prefixes that generate the highest outbound traffic volume. Throughput prefixes are sorted from highest to lowest. Delay learning measures prefixes with the highest round-trip response time (RTT) to optimize these highest delay prefixes to try to reduce the RTT for these prefixes. Delay prefixes are sorted from the highest to the lowest delay time.

OER can automatically learn two types of prefixes:

- outside prefix--An outside prefix is defined as a public IP prefix assigned outside the company. Outside prefixes are received from other networks.
- inside prefix--An inside prefix is defined as a public IP prefix assigned to a company. An inside prefix is a prefix configured within the company network.

In Cisco IOS Release 12.4(9)T, 12.2(33)SRB, and later releases, the ability to learn inside prefixes was introduced. Using BGP, OER can select inside prefixes to support best entrance selection for traffic that originates from prefixes outside an autonomous system destined for prefixes inside the autonomous system. In prior releases, only outside prefixes were supported. Company networks advertise the inside prefixes over the Internet using an Internet service provider (ISP) and receive advertisements for outside prefixes from an ISP.

Note Although OER can learn an inside prefix, OER will not try to control an inside prefix unless there is an exact match in the BGP routing information base (RIB) because OER does not advertise a new prefix to the Internet.

Automatic prefix learning is configured in OER Top Talker and Top Delay learning configuration mode. The **learn** command is used to enter this mode from OER master controller configuration mode. When

automatic prefix learning is enabled, prefixes and their delay or throughput characteristics are measured on the border routers. Performance measurements for the prefix-based traffic classes are reported to the master controller where the learned prefixes are stored in the MTC list.

Prefixes are learned on the border routers through monitoring the traffic flow using the embedded NetFlow capability. All incoming and outgoing traffic flows are monitored. The top 100 flows are learned by default, but the master controller can be configured to learn up to 2500 flows for each learn cycle. In Cisco IOS Release 12.4(20)T, 12.2(33)SRE, and later releases, the limit of 5000 prefixes that can be controlled by a master controller was removed. In Cisco IOS Release 12.4(15)T, 12.2(33)SRB, 12.2(33)SXH, and earlier releases, the master controller can control a maximum of 5000 prefixes.

The master controller can be configured to aggregate learned prefixes based on type, BGP or non-BGP (static). Prefixes can be aggregated based on the prefix length. Traffic flows are aggregated using a /24 prefix length by default. Prefix aggregation can be configured to include any subset or superset of the network, from single host route (/32) to a major network address range. For each aggregated prefix, up to five host addresses are selected to use as active probe targets. Prefix aggregation is configured with the **aggregation-type** command in OER Top Talker and Delay learning configuration mode.

Application Traffic Class Learning Using OER

In the first release of OER, Cisco IOS Release 12.3(8)T, only Layer 3 prefixes could be learned. In subsequent releases, Layer 4 options such as protocol or port numbers were added as filters to the prefix-based traffic class. The protocol and port numbers can be used to identify specific application traffic classes; protocol and port number parameters are monitored only within the context of a prefix and are not sent to the master controller database (MTC list). The prefix that carries the specific traffic is then monitored by the master controller. In Cisco IOS Release 12.4(9)T, Release 12.2(33)SRB, and later releases, application traffic class learning supports Differentiated Services Code Point (DSCP) values in addition to protocol and port numbers, and these Layer 4 options are entered in the MTC list.

Port and Protocol Based Prefix Learning by OER

In Cisco IOS Release 12.3(11)T, Release 12.2(33)SRB, and later releases, prefix learning on the basis of port numbers or protocols was introduced. This feature allows you to configure the master controller to filter the prefix-based traffic class based on the protocol number or the source or destination port number, carried by TCP or UDP traffic. This feature provides a very granular filter that can be used to further optimize prefixes learned based on throughput and delay. The traffic classes sent to the MTC list on the master controller, however, only contain the prefix information, not the protocol and port numbers.

Port and protocol based prefix learning allows you to optimize or exclude traffic streams for a specific protocol or the TCP port, UDP port, or range of port numbers. Traffic can be optimized for a specific application or protocol. Uninteresting traffic can be excluded, allowing you to focus router system resources, and reduce unnecessary CPU and memory utilization. In cases where traffic streams need to be excluded or included over ports that fall above or below a certain port number, the range of port numbers can be specified. Port and protocol prefix based learning is configured with the **protocol** command in OER Top Talker and Top Delay learning configuration mode.

For a list of IANA assigned port numbers, see the following document:

- http://www.iana.org/assignments/port-numbers

For a list of IANA assigned protocol numbers, see the following document:

- http://www.iana.org/assignments/protocol-numbers

DSCP Value, Port, and Protocol Learning by OER

In Cisco IOS Release 12.4(9)T, 12.2(33)SRB, and later releases, the ability to filter and aggregate application traffic by DSCP value, port number or protocol was introduced. Traffic classes can be defined by a combination of keys comprising of protocol, port numbers, and DSCP values. The ability to filter out traffic that is not required, and the ability to aggregate the traffic in which you are interested, was introduced. Information such as protocol, port number, and DSCP value is now sent to the master controller database in addition to the prefix information. The new functionality allows OER to both actively and passively monitor application traffic. Using new CLI and access lists, OER can be configured to automatically learn application traffic classes.

Learn List Configuration Mode

In Cisco IOS Release 12.4(15)T, a new configuration mode named learn list was introduced. Learn lists are a way to categorize learned traffic classes. In each learn list, different criteria including prefixes, application definitions, filters, and aggregation parameters for learning traffic classes can be configured.

If you are running Cisco IOS Release 12.4(15)T or a later release, please refer to the Using Performance Routing to Profile the Traffic Classes module for learn list configuration information and tasks introduced for Performance Routing. Performance Routing (PfR) is an extension of the Optimized Edge Routing (OER) technology and the commands and command modes for PfR use the oer naming convention.

OER Manual Traffic Class Configuration

OER can be manually configured to create traffic classes for monitoring and subsequent optimizing. Automatic learning generally uses a default prefix length of /24 but manual configuration allows exact prefixes to be defined. Within the manual traffic class configuration process there are two components--manually configuring prefix-based traffic classes and manually configuring application-based traffic classes, both of which are described in the following sections:

- Prefix Traffic Class Configuration Using OER, page 5
- Application Traffic Class Configuration Using OER, page 6

Prefix Traffic Class Configuration Using OER

A prefix or range of prefixes can be selected for OER monitoring by configuring an IP prefix list. The IP prefix list is then imported into the MTC list by configuring a match clause in an OER map. An OER map is similar to an IP route map. IP prefix lists are configured with the **ip prefix-list** command and OER maps are configured with the **oer-map** command in global configuration mode.

The prefix list syntax operates in a slightly different way with OER than in regular routing. The **ge** keyword is not used and the **le** keyword is used by OER to specify only an inclusive prefix. A prefix list can also be used to specify an exact prefix.

A master controller can monitor and control an exact prefix of any length including the default route. If an exact prefix is specified, OER monitors only the exact prefix.

A master controller can monitor and control an inclusive prefix using the **le** keyword and the *le-value* argument set to 32. OER monitors the configured prefix and any more specific prefixes (for example, configuring the 10.0.0.0/8 le 32 prefix would include the 10.1.0.0/16 and the 10.1.1.0/24 prefixes) over the same exit and records the information in the routing information base (RIB).

Note Use the inclusive prefix option with caution in a typical OER deployment because of the potential increase in the amount of prefixes being monitored and recorded.

An IP prefix list with a deny statement can be used to configure the master controller to exclude a prefix or prefix length for learned traffic classes. Deny prefix list sequences should be applied in the lowest OER map sequences for best performance. In Cisco IOS Release 12.4(9)T, 12.2(33)SRB, and later releases, the master controller can be configured to tell border routers to filter out uninteresting traffic using an access list.

Note IP prefix lists with deny statements can be applied only to learned traffic classes.

Two types of prefix can be manually configured for OER monitoring using an IP prefix list:

- outside prefix--An outside prefix is defined as a public IP prefix assigned outside the company. Outside prefixes are received from other networks.
- inside prefix--An inside prefix is defined is defined as a public IP prefix assigned to a company. An inside prefix is a prefix configured within the company network.

In Cisco IOS Release 12.4(9)T, 12.2(33)SRB, and later releases, the ability to manually configure inside prefixes was introduced. Using BGP, OER can be configured to select inside prefixes to support best entrance selection for traffic that originates from prefixes outside an autonomous system destined for prefixes inside the autonomous system. In prior releases, only outside prefixes were supported. Company networks advertise the inside prefixes over the Internet using an Internet service provider (ISP) and receive advertisements for outside prefixes from an ISP.

Note Although an inside prefix can be manually configured for OER monitoring, OER will not try to control an inside prefix unless there is an exact match in the BGP routing information base (RIB) because OER does not advertise a new prefix to the Internet.

Application Traffic Class Configuration Using OER

In the first release of OER, Cisco IOS Release 12.3(8)T, only Layer 3 prefixes could be manually configured during the OER profile phase. In Cisco IOS Release 12.4(2)T, 12.2(33)SRB, and later releases, support for OER application-aware routing for policy-based routing (PBR) was introduced. Application-aware routing allows the selection of traffic for specific applications based on values in the IP packet header, other than the Layer 3 destination address through a named extended IP access control list (ACL). Only named extended ACLs are supported. The extended ACL is configured with a permit statement and then referenced in an OER map.

Note On Cisco 6500 Series Switches there is a limit of 15,000 ACL entries for each traffic class filter.

The protocol and port numbers can be used to identify specific application traffic classes, but protocol and port number parameters are monitored only within the context of a prefix, and are not sent to the MTC list. Only the prefix that carries the specific application traffic is profiled by the master controller. With application-aware routing support, active monitoring of application traffic was supported. Passive

monitoring of application traffic was introduced in Cisco IOS Release 12.4(9)T, 12.2(33)SRB, and later releases, with application traffic class configuration support of the profiling of DSCP values as well as protocol and port numbers. DSCP values, port numbers, and protocols in addition to prefixes, are all now stored in the MTC list.

In Cisco IOS Release 12.4(15)T, new static application mapping was introduced under OER map configuration mode to simplify the configuration of traffic classes. If you are running Cisco IOS Release 12.4(15)T or a later release, please refer to the Using Performance Routing to Profile the Traffic Classes module for static application mapping configuration information and tasks introduced for Performance Routing. Performance Routing (PfR) is an extension of the Optimized Edge Routing (OER) technology and the commands and command modes for PfR use the oer naming convention.

How to Configure OER to Profile the Traffic Classes

An OER master controller can be configured to automatically learn the traffic classes, or the traffic classes can be manually configured. Two types of traffic classes--to be automatically learned or manually configured--can be profiled:

- Traffic classes based on destination prefixes
- Traffic classes representing custom application definitions using access lists

Note In Cisco IOS Release 12.4(15)T, the introduction of learn lists allows traffic classes that are automatically learned by OER to be categorized into separate learn lists to which different OER policies can be applied. If you are running Cisco IOS Release 12.4(15)T or a later release, please refer to the Using Performance Routing to Profile the Traffic Classes module for learn list configuration information and tasks introduced for Performance Routing. Performance Routing (PfR) is an extension of the Optimized Edge Routing (OER) technology and the commands and command modes for PfR use the oer naming convention.

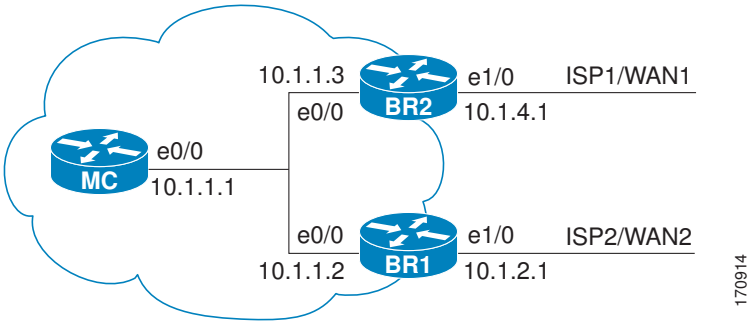
One or more of the following tasks may be performed:

- Configuring OER to Automatically Learn Prefix-Based Traffic Classes, page 8
- Configuring OER to Automatically Learn Traffic Classes Using Inside Prefixes, page 11
- Configuring OER to Automatically Learn Prefix-Based Traffic Classes Using Protocol or Port Number, page 14
- Specifying the Flow Keys for Automatic Learning of Application Traffic Classes, page 17
- Creating an Access List to Specify a Filter for Automatically Learned Application Traffic, page 20
- Displaying Application Traffic Flow Information on a Border Router, page 28
- Manually Selecting Prefixes for OER Monitoring, page 30
- Manually Selecting Inside Prefixes for OER Monitoring, page 32
- Manually Selecting Traffic Classes Using Prefix Protocol Port and DSCP Value, page 34

Configuring OER to Automatically Learn Prefix-Based Traffic Classes

Perform this task to configure an OER master controller to automatically learn prefixes to be used as traffic classes to be entered in the MTC list. This task is performed on the master controller shown in the figure below.

Figure 2 Network Diagram of OER Master Controller and Border Routers



The **learn** command is entered in OER master controller configuration mode and is required to enter OER Top Talker and Top Delay configuration mode. This task configures prefix learning based on the highest outbound throughput or the highest delay time, and one or both of these parameters must be specified. Optional configuration parameters such as learning period timers, maximum number of prefixes, and an expiration time for MTC list entries are also shown.

Note If any of the border routers is a Cisco Catalyst 6500 switch or a Cisco 7600 series router, there are some hardware constraints and the master controller will set the monitoring mode to special where only the throughput method of learning is used to profile the traffic classes. Do not configure Step 5 (the **delay** command) of this task if any of the border routers is a Cisco Catalyst 6500 switch or a Cisco 7600 series router. For more details about the special monitoring mode, see the Measuring the Traffic Class Performance and Link Utilization Using OER module.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **oer master**
4. **learn**
5. **delay**
6. **throughput**
7. **aggregation-type** {**bgp non-bgp prefix-length**} *prefix-mask*
8. **monitor-period** *minutes*
9. **periodic-interval** *minutes*
10. **prefixes** *number*
11. **expire after session** *number* **time** *minutes*
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**oer master** **Example:** `Router(config)# oer master`	Enters OER master controller configuration mode to configure a Cisco router as a master controller and to configure master controller policy and timer settings.
Step 4	**learn** **Example:** `Router(config-oer-mc)# learn`	Enters OER Top Talker and Top Delay learning configuration mode to configure prefix learning and timers.
Step 5	**delay** **Example:** `Router(config-oer-mc-learn)# delay`	Enables prefix learning based on the highest delay time. • *Top Delay* prefixes are sorted from the highest to lowest delay time. • The example configures prefix learning based on the highest delay. **Note** To configure OER learning you must specify either the **delay** command, the **throughput** command, or both commands.
Step 6	**throughput** **Example:** `Router(config-oer-mc-learn)# throughput`	Configures the master controller to learn the top prefixes based on the highest outbound throughput. • When this command is enabled, the master controller will learn the top prefixes across all border routers according to the highest outbound throughput. • The example configures a master controller to learn the top prefixes based on highest outbound throughput.

	Command or Action	Purpose
Step 7	**aggregation-type** {**bgp non-bgp prefix-length**} *prefix-mask* **Example:** `Router(config-oer-mc-learn)# aggregation-type bgp`	(Optional) Configures a master controller to aggregate learned prefixes based on traffic flow type. • The **bgp** keyword configures prefix aggregation based on entries in the BGP routing table. This keyword is used if BGP peering is enabled in the network. • The **non-bgp** keyword configures learned prefix aggregation based on static routes. Entries in the BGP routing table are ignored when this keyword is entered. • The **prefix-length** keyword configures aggregation based on the specified prefix length. The range of values that can be configured for this argument is a prefix mask from 1 to 32. • If this command is not specified, the default aggregation is performed based on a /24 prefix length. • The example configures BGP prefix aggregation.
Step 8	**monitor-period** *minutes* **Example:** `Router(config-oer-mc-learn)# monitor-period 10`	(Optional) Sets the time period that an OER master controller learns traffic flows. • The default learning period is 5 minutes. • The length of time between monitoring periods is configured with the **periodic-interval** command. • The number of prefixes that are learned is configured with the **prefixes** command. • The example sets the length of each monitoring period to 10 minutes.
Step 9	**periodic-interval** *minutes* **Example:** `Router(config-oer-mc-learn)# periodic-interval 20`	(Optional) Sets the time interval between prefix learning periods. • By default, the interval between prefix learning periods is 120 minutes. • The example sets the time interval between monitoring periods to 20 minutes.
Step 10	**prefixes** *number* **Example:** `Router(config-oer-mc-learn)# prefixes 200`	(Optional) Sets the number of prefixes that the master controller will learn during the monitoring period. • By default, the top 100 traffic flows are learned. • The example configures a master controller to learn 200 prefixes during each monitoring period.

	Command or Action	Purpose
Step 11	**expire after session** *number* **time** *minutes* **Example:** `Router(config-oer-mc-learn)# expire after session 100`	(Optional) Sets the length of time that learned prefixes are kept in the central policy database. • The **session** keyword configures learned prefixes to be removed after the specified number of monitoring periods have occurred. • The **time** keyword configures learned prefixes to be removed after the specified time period. The time value is entered in minutes. • The example configures learned prefixes to be removed after 100 monitoring periods.
Step 12	**end** **Example:** `Router(config-oer-mc)# end`	Exits OER Top Talker and Top Delay learning configuration mode, and returns to privileged EXEC mode.

• What to Do Next, page 11

What to Do Next

This section shows how to configure automatic prefix learning. To configure specific prefixes for OER monitoring and optimization, see the OER Traffic Class Profiling section.

Configuring OER to Automatically Learn Traffic Classes Using Inside Prefixes

In Cisco IOS Release 12.4(9)T, 12.2(33)SRB, and later releases, the OER BGP inbound optimization feature introduced the ability to automatically learn inside prefixes to support best entrance selection for traffic that originates from prefixes outside an autonomous system destined for prefixes inside the autonomous system.

Perform this task to configure an OER master controller to automatically learn inside prefixes to be used as traffic classes to be entered in the MTC list. This task is configured at the master controller and introduces the **inside bgp** command used in OER Top Talker and Top Delay configuration mode. This task configures automatic prefix learning of the inside prefixes (prefixes within the network). Optional configuration parameters such as learning period timers, maximum number of prefixes, and an expiration time for MTC list entries are also shown.

- Before configuring this task, BGP peering for internal and external BGP neighbors must be configured.
- This task requires Cisco IOS Release 12.4(9)T, 12.2(33)SRB, or later release to be running on the master controller and border routers.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **oer master**
4. **learn**
5. **inside bgp**
6. **monitor-period** *minutes*
7. **periodic-interval** *minutes*
8. **prefixes** *number*
9. **expire after session** *number* | **time** *minutes*
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**oer master** **Example:** `Router(config)# oer master`	Enters OER master controller configuration mode to configure a router as a master controller and to configure global operations and policies.
Step 4	**learn** **Example:** `Router(config-oer-mc)# learn`	Enters OER Top Talker and Top Delay learning configuration mode to configure prefix learning policies and timers.
Step 5	**inside bgp** **Example:** `Router(config-oer-mc-learn)# inside bgp`	Learns prefixes inside the network.

	Command or Action	Purpose
Step 6	**monitor-period** *minutes* **Example:** `Router(config-oer-mc-learn)# monitor-period 10`	(Optional) Sets the time period that an OER master controller learns traffic flows. • The default learning period is 5 minutes. • The length of time between monitoring periods is configured with the **periodic-interval** command. • The number of prefixes that are learned is configured with the **prefixes** command. • The example sets the length of each monitoring period to 10 minutes.
Step 7	**periodic-interval** *minutes* **Example:** `Router(config-oer-mc-learn)# periodic-interval 20`	(Optional) Sets the time interval between prefix learning periods. • By default, the interval between prefix learning periods is 120 minutes. • The example sets the time interval between monitoring periods to 20 minutes.
Step 8	**prefixes** *number* **Example:** `Router(config-oer-mc-learn)# prefixes 200`	(Optional) Sets the number of prefixes that the master controller will learn during the monitoring period. • By default, the top 100 traffic flows are learned. • The example configures a master controller to learn 200 prefixes during each monitoring period.
Step 9	**expire after session** *number* | **time** *minutes* **Example:** `Router(config-oer-mc-learn)# expire after session 100`	(Optional) Sets the length of time that learned prefixes are kept in the central policy database. • The **session** keyword configures learned prefixes to be removed after the specified number of monitoring periods have occurred. • The **time** keyword configures learned prefixes to be removed after the specified time period. The time value is entered in minutes. • The example configures learned prefixes to be removed after 100 monitoring periods.
Step 10	**end** **Example:** `Router(config-oer-mc-learn)# end`	Exits OER Top Talker and Top Delay learning configuration mode, and enters privileged EXEC mode.

• What to Do Next, page 13

What to Do Next

This section shows how to configure automatic prefix learning for inside prefixes. To configure specific inside prefixes for OER monitoring and optimization, see the Manually Selecting Inside Prefixes for OER Monitoring.

Configuring OER to Automatically Learn Prefix-Based Traffic Classes Using Protocol or Port Number

Perform this task to configure an OER master controller to learn traffic classes to be entered in the MTC list based on prefixes but filtered by the protocol or port number. This task is performed on a master controller. The **learn** command is entered in OER master controller configuration mode and is required to enter OER Top Talker and Top Delay configuration mode. This task configures prefix learning based on the highest outbound throughput or the highest delay time and one or both of these parameters must be specified. After the prefix has been learned, a protocol or port number can be specified to create a subset of traffic classes. Optional configuration parameters such as learning period timers, the maximum number of prefixes, and an expiration time for MTC list entries are also shown.

This task requires Cisco IOS Release 12.3(11)T, 12.2(33)SRB, or later release, to be running on the master controller and border routers.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **oer master**
4. **learn**
5. **delay**
6. **throughput**
7. **aggregation-type** {**bgp non-bgp prefix-length**} *prefix-mask*
8. **monitor-period** *minutes*
9. **periodic-interval** *minutes*
10. **prefixes** *number*
11. **expire after** {**session** *number* | **time** *minutes*}
12. **protocol** {*protocol-number* | **tcp** | **udp**} [**port** *port-number* | **gt** *port-number* | **lt** *port-number* **range** *lower-number upper-number*][**dst** | **src**]
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**oer master** **Example:** `Router(config)# oer master`	Enters OER master controller configuration mode to configure a Cisco router as a master controller and to configure master controller policy and timer settings.
Step 4	**learn** **Example:** `Router(config-oer-mc)# learn`	Enters OER Top Talker and Top Delay learning configuration mode to configure prefix learning policies and timers.
Step 5	**delay** **Example:** `Router(config-oer-mc-learn)# delay`	Enables prefix learning based on the highest delay time. • *Top Delay* prefixes are sorted from the highest to lowest delay time. • The example configures prefix learning based on the highest delay. **Note** To configure OER learning you must specify either the **delay** command, the **throughput** command, or both commands.
Step 6	**throughput** **Example:** `Router(config-oer-mc-learn)# throughput`	Configures the master controller to learn the top prefixes based on the highest outbound throughput. • When this command is enabled, the master controller will learn the top prefixes across all border routers according to the highest outbound throughput. • The example configures a master controller to learn the top prefixes based on highest outbound throughput.
Step 7	**aggregation-type** {**bgp non-bgp prefix-length**} *prefix-mask* **Example:** `Router(config-oer-mc-learn)# aggregation-type bgp`	(Optional) Configures a master controller to aggregate learned prefixes based on traffic flow type. • The **bgp** keyword configures prefix aggregation based on entries in the BGP routing table. This keyword is used if BGP peering is enabled in the network. • The **non-bgp** keyword configures learned prefix aggregation based on static routes. Entries in the BGP routing table are ignored when this keyword is entered. • The **prefix-length** keyword configures aggregation based on the specified prefix length. The range of values that can be configured for this argument is a prefix mask from 1 to 32. • If this command is not specified, the default aggregation is performed based on a /24 prefix length. • Up to five host addresses are learned for active monitoring when a prefix is aggregated. • The example configures BGP prefix aggregation.

	Command or Action	Purpose
Step 8	**monitor-period** *minutes* **Example:** `Router(config-oer-mc-learn)# monitor-period 10`	(Optional) Sets the time period that an OER master controller learns traffic flows. • The default learning period is 5 minutes. • The length of time between monitoring periods is configured with the **periodic-interval** command. • The number of prefixes that are learned is configured with the **prefixes** command. • The example sets the length of each monitoring period to 10 minutes.
Step 9	**periodic-interval** *minutes* **Example:** `Router(config-oer-mc-learn)# periodic-interval 20`	(Optional) Sets the time interval between prefix learning periods. • By default, the interval between prefix learning periods is 120 minutes. • The example sets the time interval between monitoring periods to 20 minutes.
Step 10	**prefixes** *number* **Example:** `Router(config-oer-mc-learn)# prefixes 200`	(Optional) Sets the number of prefixes that the master controller will learn during the monitoring period. • By default, the top 100 traffic flows are learned. • The example configures a master controller to learn 200 prefixes during each monitoring period.
Step 11	**expire after** {**session** *number* | **time** *minutes*} **Example:** `Router(config-oer-mc-learn)# expire after session 100`	(Optional) Sets the length of time that learned prefixes are kept in the central policy database. • The **session** keyword configures learned prefixes to be removed after the specified number of monitoring periods have occurred. • The **time** keyword configures learned prefixes to be removed after the specified time period. The time value is entered in minutes. • The example configures learned prefixes to be removed after 100 monitoring periods.

	Command or Action	Purpose
Step 12	**protocol** {*protocol-number* | **tcp** | **udp**} [**port** *port-number* | **gt** *port-number* | **lt** *port-number* **range** *lower-number upper-number*][**dst** | **src**] **Example:** `Router(config-oer-mc-learn)# protocol tcp port range 49542 49478`	Configures the master controller to learn prefixes based on a protocol number, TCP or UDP port number, or a range of port numbers. • Filtering based on a specific protocol is configured with the *protocol-number* argument. • TCP or UDP based filtering is enabled by configuring the **tcp** or **udp** keyword. • Port based filtering is enabled by configuring the **port** keyword. Port number ranges can be filtered based on greater-than or equal-to and less-than or equal-to filtering, or can be filtered by specifying a starting and ending port numbers with the **range**keyword. • Destination or source port-based filtering is enabled by configuring the **dst** or **src** keywords. • The example configures a master controller to learn prefixes from a database during each monitoring period. The database traffic is identified by a range of port numbers.
Step 13	**end** **Example:** `Router(config-oer-mc)# end`	Exits OER Top Talker and Top Delay learning configuration mode, and returns to privileged EXEC mode.

• What to Do Next, page 17

What to Do Next

This section shows how to configure automatic prefix-based traffic class learning using protocol or port number. To configure specific prefix-based traffic classes using protocol or port numbers for OER monitoring and optimization, see Manually Selecting Traffic Classes Using Prefix Protocol Port and DSCP Value.

Specifying the Flow Keys for Automatic Learning of Application Traffic Classes

Perform this task at the master controller to define the application traffic flow fields that OER can use to automatically learn traffic classes to be entered in the MTC list. In Cisco IOS Release 12.4(9)T, 12.2(33)SRB, and later releases, traffic class commands were introduced to help define the application traffic classes. The traffic class commands can be used in the following situations:

- You can use the filter and aggregation traffic class commands with the traffic class keys. Traffic class keys are specified, but they will be used only if the traffic class aggregation access list does not have any matches. In this situation, some knowledge of the prefixes that OER will learn is presumed.
- You can also use this task without the traffic class commands that use the filter and aggregation access lists, if you do not want to filter or aggregate any traffic classes. In this situation, no knowledge of the prefixes is presumed and only the traffic class command that specifies the keys is used.

In Cisco IOS Release 12.4(9)T and 12.2(33)SRB the ability to learn traffic using protocol, port number, and DSCP value (in addition to prefix) was introduced. Specifying the protocol, ports, and DSCP value

allows application traffic to be identified in more detail. In this task, only traffic class keys are specified for voice traffic. The voice application traffic is identified by the UDP protocol, a DSCP value of ef, and port numbers in the range from 3000 to 4000. The master controller is also configured to learn the top prefixes based on highest outbound throughput for the specified traffic and the resulting traffic classes are added to the OER application database to be passively and actively monitored.

To display information about the traffic classes learned by OER use Displaying Application Traffic Flow Information on a Border Router.

This task requires the master controller and border routers to be running Cisco IOS Release 12.4(9)T, 12.2(33)SRB, or later releases.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **oer master**
4. **learn**
5. **aggregation-type** {**bgp non-bgp prefix-length**} *prefix-mask*
6. **throughput**
7. **monitor-period** *minutes*
8. **periodic-interval** *minutes*
9. **prefixes** *number*
10. **traffic-class keys** [[**default**] | [**sport**] [**dport**] [**dscp**] [**protocol**]]
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**oer master** **Example:** `Router(config)# oer master`	Enters OER master controller configuration mode to configure a Cisco router as a master controller and to configure master controller policy and timer settings.

	Command or Action	Purpose
Step 4	**learn** **Example:** `Router(config-oer-mc)# learn`	Enters OER Top Talker and Top Delay learning configuration mode to configure prefix learning policies and timers.
Step 5	**aggregation-type** {**bgp non-bgp prefix-length**} *prefix-mask* **Example:** `Router(config-oer-mc-learn)# aggregation-type prefix-length 24`	(Optional) Configures a master controller to aggregate learned prefixes based on traffic flow type. • The **bgp** keyword configures prefix aggregation based on entries in the BGP routing table. This keyword is used if BGP peering is enabled in the network. • The **non-bgp** keyword configures learned prefix aggregation based on static routes. Entries in the BGP routing table are ignored when this keyword is entered. • The **prefix-length** keyword configures aggregation based on the specified prefix length. The range of values that can be configured for this argument is a prefix mask from 1 to 32. • If this command is not specified, the default aggregation is performed based on a /24 prefix length. • The example configures prefix length aggregation.
Step 6	**throughput** **Example:** `Router(config-oer-mc-learn)# throughput`	Configures the master controller to learn the top prefixes based on the highest outbound throughput. • When this command is enabled, the master controller will learn the top prefixes across all border routers according to the highest outbound throughput. • The example configures a master controller to learn the top prefixes based on highest outbound throughput.
Step 7	**monitor-period** *minutes* **Example:** `Router(config-oer-mc-learn)# monitor-period 10`	(Optional) Sets the time period that an OER master controller learns traffic flows. • The default learning period is 5 minutes. • The length of time between monitoring periods is configured with the **periodic-interval** command. • The number of prefixes that are learned is configured with the **prefixes** command. • The example sets the length of each monitoring period to 10 minutes.
Step 8	**periodic-interval** *minutes* **Example:** `Router(config-oer-mc-learn)# periodic-interval 20`	(Optional) Sets the time interval between prefix learning periods. • By default, the interval between prefix learning periods is 120 minutes. • The example sets the time interval between monitoring periods to 20 minutes.

	Command or Action	Purpose
Step 9	**prefixes** *number* **Example:** `Router(config-oer-mc-learn)# prefixes 200`	(Optional) Sets the number of prefixes that the master controller will learn during the monitoring period. • By default, the top 100 traffic flows are learned. • The example configures a master controller to learn 200 prefixes during each monitoring period.
Step 10	**traffic-class keys** [[**default**] | [**sport**] [**dport**] [**dscp**] [**protocol**]] **Example:** `Router(config-oer-mc-learn)# traffic-class keys dport dscp protocol`	Specifies a key list used by the border router to aggregate the traffic flows into the learn aggregation cache. • Traffic class keys are used when there is no traffic class aggregation access list or if the traffic class aggregation access list does not have any matches. • The example specifies a key list of destination port, dscp value, and protocol.
Step 11	**end** **Example:** `Router(config-oer-mc-learn)# end`	Exits OER Top Talker and Top Delay learning configuration mode, and returns to privileged EXEC mode.

Creating an Access List to Specify a Filter for Automatically Learned Application Traffic

Perform this task at the master controller to create an access list to filter specific application traffic for OER monitoring. In Cisco IOS Release 12.4(9)T and 12.2(33)SRB the ability to learn traffic using protocol, port number, and DSCP value (in addition to prefix) was introduced. Specifying the protocol, ports, and DSCP value allows application traffic to be identified in more detail.

In the Specifying the Flow Keys for Automatic Learning of Application Traffic Classes task, traffic keys were used to identify application traffic because no knowledge of any of the prefixes was assumed. If you know some prefixes that you want to exclude, then you can use this task to create an access list and filter out unwanted traffic. In this example for Voice traffic, the access list, VOICE_FILTER_LIST, configures OER to identify all UDP traffic from any source to a destination prefix of 10.1.0.0/16 with a DSCP value of ef that represents voice traffic. The access list is applied using a traffic class command that filters out unwanted traffic. The master controller is also configured to learn the top prefixes based on highest outbound throughput for the filtered traffic and the resulting traffic classes are added to the OER application database to be passively and actively monitored.

To display information about the traffic classes learned by OER use the Displaying Application Traffic Flow Information on a Border Router task.

This task requires the master controller and border routers to be running Cisco IOS Release 12.4(9)T, 12.2(33)SRB, or later releases.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list** {**standard** | **extended**} *access-list-name*
4. [*sequence-number*] **permit udp** *source source-wildcard* [*operator* [*port*]] *destination destination-wildcard* [*operator* [*port*]] [**dscp** *dscp-value*]
5. **exit**
6. **oer master**
7. **learn**
8. **aggregation-type** {**bgp non-bgp prefix-length**} *prefix-mask*
9. **throughput**
10. **monitor-period** *minutes*
11. **periodic-interval** *minutes*
12. **prefixes** *number*
13. **traffic-class filter access-list** *access-list-name*
14. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip access-list** {**standard** | **extended**} *access-list-name* **Example:** `Router(config)# ip access-list extended VOICE_FILTER_LIST`	Defines an IP access list by name. • OER supports only named access lists. • The example creates an extended IP access list named VOICE_FILTER_LIST.

	Command or Action	Purpose
Step 4	[*sequence-number*] **permit udp** *source source-wildcard* [*operator* [*port*]] *destination destination-wildcard* [*operator* [*port*]] [**dscp** *dscp-value*] **Example:** `Router(config-ext-nacl)# permit udp any 10.1.0.0 0.0.255.255 dscp ef`	Sets conditions to allow a packet to pass a named IP access list. • The example is configured to identify all UDP traffic from any source to a destination prefix of 10.1.0.0/16 where the DSCP bit is set to ef. This specific UDP traffic is to be optimized. **Note** Only the syntax applicable to this task is shown. For more details, see the *Cisco IOS IP Application Services Command Reference*.
Step 5	**exit** **Example:** `Router(config-ext-nacl)# exit`	(Optional) Exits extended access list configuration mode and returns to global configuration mode.
Step 6	**oer master** **Example:** `Router(config)# oer master`	Enters OER master controller configuration mode to configure a Cisco router as a master controller and to configure master controller policy and timer settings.
Step 7	**learn** **Example:** `Router(config-oer-mc)# learn`	Enters OER Top Talker and Top Delay learning configuration mode to configure prefix learning policies and timers.
Step 8	**aggregation-type** {**bgp non-bgp prefix-length**} *prefix-mask* **Example:** `Router(config-oer-mc-learn)# aggregation-type prefix-length 24`	(Optional) Configures a master controller to aggregate learned prefixes based on traffic flow type. • The **bgp** keyword configures prefix aggregation based on entries in the BGP routing table. This keyword is used if BGP peering is enabled in the network. • The **non-bgp** keyword configures learned prefix aggregation based on static routes. Entries in the BGP routing table are ignored when this keyword is entered. • The **prefix-length** keyword configures aggregation based on the specified prefix length. The range of values that can be configured for this argument is a prefix mask from 1 to 32. • If this command is not specified, the default aggregation is performed based on a /24 prefix length. • The example configures prefix length aggregation.

	Command or Action	Purpose
Step 9	**throughput** **Example:** `Router(config-oer-mc-learn)# throughput`	Configures the master controller to learn the top prefixes based on the highest outbound throughput. • When this command is enabled, the master controller will learn the top prefixes across all border routers according to the highest outbound throughput. • The example configures a master controller to learn the top prefixes based on highest outbound throughput.
Step 10	**monitor-period** *minutes* **Example:** `Router(config-oer-mc-learn)# monitor-period 10`	(Optional) Sets the time period that an OER master controller learns traffic flows. • The default learning period is 5 minutes. • The length of time between monitoring periods is configured with the **periodic-interval** command. • The number of prefixes that are learned is configured with the **prefixes** command. • The example sets the length of each monitoring period to 10 minutes.
Step 11	**periodic-interval** *minutes* **Example:** `Router(config-oer-mc-learn)# periodic-interval 20`	(Optional) Sets the time interval between prefix learning periods. • By default, the interval between prefix learning periods is 120 minutes. • The example sets the time interval between monitoring periods to 20 minutes.
Step 12	**prefixes** *number* **Example:** `Router(config-oer-mc-learn)# prefixes 200`	(Optional) Sets the number of prefixes that the master controller will learn during the monitoring period. • By default, the top 100 traffic flows are learned. • The example configures a master controller to learn 200 prefixes during each monitoring period.
Step 13	**traffic-class filter access-list** *access-list-name* **Example:** `Router(config-oer-mc-learn)# traffic-class filter access-list VOICE_FILTER_LIST`	Supports filtering of traffic classes during OER passive monitoring by using an extended access list. • The example configures learned prefixes to be filtered using the access list named VOICE_FILTER_LIST that was created in Step 3 of this task.
Step 14	**end** **Example:** `Router(config-oer-mc-learn)# end`	Exits OER Top Talker and Top Delay learning configuration mode, and returns to privileged EXEC mode.

• Creating an Access List to Specify Aggregation Criteria for Automatically Learned Application Traffic, page 24

Creating an Access List to Specify Aggregation Criteria for Automatically Learned Application Traffic

Perform this task at the master controller to create an access list to aggregate learned application traffic for OER monitoring. In Cisco IOS Release 12.4(9)T and 12.2(33)SRB the ability to learn traffic using protocol, port number, and DSCP value (in addition to prefix) was introduced. Specifying the protocol, ports, and DSCP value allows application traffic to be identified in more detail.

In the Creating an Access List to Specify a Filter for Automatically Learned Application Traffic task, the application traffic was filtered to profile traffic for a specific destination prefix, but in this task, the application traffic is being aggregated for a range of destination ports. In this example, the access list, VOICE_AGG_LIST is configured to aggregate traffic with a destination port in the range from 3000 to 4000 and with a DSCP value of ef. This UDP traffic represents voice traffic and OER will create traffic classes based on the specified port number range and DSCP value. In this task, the master controller is also configured to learn the top prefixes based on highest outbound throughput for the aggregated traffic and the resulting traffic classes are added to the OER application database to be passively and actively monitored.

The last step in this task is an optional step to review the configuration on the OER master controller. To display more information about the traffic classes learned by OER use the Displaying Application Traffic Flow Information on a Border Router task.

This task requires the master controller and border routers to be running Cisco IOS Release 12.4(9)T, 12.2(33)SRB, or later releases.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list** {**standard** | **extended**} *access-list-name*
4. [*sequence-number*] **permit udp** *source source-wildcard* [*operator* [*port*]] *destination destination-wildcard* [*operator* [*port*]] [**dscp** *dscp-value*]
5. **exit**
6. **oer master**
7. **learn**
8. **aggregation-type** {**bgp non-bgp prefix-length**} *prefix-mask*
9. **throughput**
10. **monitor-period** *minutes*
11. **periodic-interval** *minutes*
12. **prefixes** *number*
13. **traffic-class aggregate access-list** *access-list-name*
14. **end**
15. **show oer master**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip access-list** {**standard** | **extended**} *access-list-name* **Example:** `Router(config)# ip access-list extended VOICE_AGG_LIST`	Defines an IP access list by name. • OER supports only named access lists. • The example creates an extended IP access list named VOICE_AGG_LIST.
Step 4	[*sequence-number*] **permit udp** *source source-wildcard* [*operator* [*port*]] *destination destination-wildcard* [*operator* [*port*]] [**dscp** *dscp-value*] **Example:** `Router(config-ext-nacl)# permit udp any any range 3000 4000 dscp ef`	Sets conditions to allow a packet to pass a named IP access list. • The example is configured to identify all UDP traffic ranging from a destination port number of 3000 to 4000 from any source where the DSCP bit is set to ef. This specific UDP traffic is to be optimized. **Note** Only the syntax applicable to this task is shown. For more details, see the *Cisco IOS IP Application Services Command Reference*.
Step 5	**exit** **Example:** `Router(config-ext-nacl)# exit`	(Optional) Exits extended access list configuration mode and returns to global configuration mode.
Step 6	**oer master** **Example:** `Router(config)# oer master`	Enters OER master controller configuration mode to configure a Cisco router as a master controller and to configure master controller policy and timer settings.

	Command or Action	Purpose
Step 7	**learn** **Example:** `Router(config-oer-mc)# learn`	Enters OER Top Talker and Top Delay learning configuration mode to configure prefix learning policies and timers.
Step 8	**aggregation-type** {**bgp non-bgp prefix-length**} *prefix-mask* **Example:** `Router(config-oer-mc-learn)# aggregation-type prefix-length 24`	(Optional) Configures a master controller to aggregate learned prefixes based on traffic flow type. • The **bgp** keyword configures prefix aggregation based on entries in the BGP routing table. This keyword is used if BGP peering is enabled in the network. • The **non-bgp** keyword configures learned prefix aggregation based on static routes. Entries in the BGP routing table are ignored when this keyword is entered. • The **prefix-length** keyword configures aggregation based on the specified prefix length. The range of values that can be configured for this argument is a prefix mask from 1 to 32. • If this command is not specified, the default aggregation is performed based on a /24 prefix length. • The example configures prefix length aggregation.
Step 9	**throughput** **Example:** `Router(config-oer-mc-learn)# throughput`	Configures the master controller to learn the top prefixes based on the highest outbound throughput. • When this command is enabled, the master controller will learn the top prefixes across all border routers according to the highest outbound throughput. • The example configures a master controller to learn the top prefixes based on highest outbound throughput.
Step 10	**monitor-period** *minutes* **Example:** `Router(config-oer-mc-learn)# monitor-period 10`	(Optional) Sets the time period that an OER master controller learns traffic flows. • The default learning period is 5 minutes. • The length of time between monitoring periods is configured with the **periodic-interval** command. • The number of prefixes that are learned is configured with the **prefixes** command. • The example sets the length of each monitoring period to 10 minutes.
Step 11	**periodic-interval** *minutes* **Example:** `Router(config-oer-mc-learn)# periodic-interval 20`	(Optional) Sets the time interval between prefix learning periods. • By default, the interval between prefix learning periods is 120 minutes. • The example sets the time interval between monitoring periods to 20 minutes.

	Command or Action	Purpose
Step 12	**prefixes** *number* **Example:** `Router(config-oer-mc-learn)# prefixes 200`	(Optional) Sets the number of prefixes that the master controller will learn during the monitoring period. • By default, the top 100 traffic flows are learned. • The example configures a master controller to learn 200 prefixes during each monitoring period.
Step 13	**traffic-class aggregate access-list** *access-list-name* **Example:** `Router(config-oer-mc-learn)# traffic-class aggregate access-list VOICE_AGG_LIST`	Supports aggregation of traffic classes during OER passive monitoring by using an extended access list. • The example configures learned prefixes to be aggregated using the access list named VOICE_AGG_LIST that was created in Step 3 of this task.
Step 14	**end** **Example:** `Router(config-oer-mc-learn)# end`	Exits OER Top Talker and Top Delay learning configuration mode, and returns to privileged EXEC mode.
Step 15	**show oer master** **Example:** `Router# show oer master`	(Optional) Displays information about the status of the OER-managed network; the output includes information about the master controller, the border routers, OER managed interfaces, and default and user-defined policy settings.

Examples

The following example output for the **show oer master** command displays the additional configuration for the traffic class aggregation, filters, and key list under the Learn Settings section.

```
Router# show oer master
OER state: ENABLED and ACTIVE
 Conn Status: SUCCESS, PORT: 7777
  Version: 2.0
  Number of Border routers: 2
  Number of Exits: 2
  Number of monitored prefixes: 0 (max 5000)
  Max prefixes: total 5000 learn 2500
  Prefix count: total 0, learn 0, cfg 0
Border           Status   UP/DOWN             AuthFail  Version
1.1.1.2          ACTIVE   UP       00:18:57          0  2.0
1.1.1.1          ACTIVE   UP       00:18:58          0  2.0
Global Settings:
  max-range-utilization percent 20 recv 20
  mode route metric bgp local-pref 5000
  mode route metric static tag 5000
  trace probe delay 1000
  logging
Default Policy Settings:
  backoff 180 200 180
  delay relative 50
  holddown 300
```

```
  periodic 0
  probe frequency 56
  mode route control
  mode monitor active
 mode select-exit good
  loss relative 10
  jitter threshold 20
  mos threshold 3.60 percent 30
  unreachable relative 50
  resolve delay priority 11 variance 20
  resolve utilization priority 12 variance 20
 *tag 0
Learn Settings:
  current state : STARTED
  time remaining in current state : 70 seconds
  throughput
  no delay
  no inside bgp
  traffic-class filter access-list voice-filter-acl <----
  traffic-class aggregate access-list voice-agg-acl <----
  traffic-class keys protocol dscp dport  <----
  no protocol
  monitor-period 2
  periodic-interval 1
  aggregation-type prefix-length 24
  prefixes 10
  expire after time 720
```

Displaying Application Traffic Flow Information on a Border Router

Perform this task to display application traffic flow information. These commands are entered on a border router through which the application traffic is flowing. The commands can be entered in any order. Keywords in Step 2 and Step 4 require the border router to be running Cisco IOS Release 12.4(9)T, 12.2(33)SRB, 12.2(33)SXH, or later releases.

This task requires the master controller and border routers to be running Cisco IOS Release 12.4(9)T, 12.2(33)SRB, or later releases.

SUMMARY STEPS

1. **enable**
2. **show oer border passive learn**
3. **show ip cache verbose flow**
4. **show oer border passive cache** {**learned** | **prefix**} [**applications]**

DETAILED STEPS

Step 1 **enable**

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Router> enable
```

Step 2 **show oer border passive learn**

This command is used to display traffic class filter and aggregation ACL information. The following example displays the voice application filter, aggregation, and keys information configured in the first three tasks under the Specifying the Flow Keys for Automatic Learning of Application Traffic Classes task.

Example:

```
Router# show oer border passive learn
OER Border Learn Configuration :
    State is enabled
    Measurement type: throughput, Duration: 2 min
    Aggregation type: prefix-length, Prefix length: 24
    No port protocol config
 Traffic Class Filter List:
   List: SrcPrefix       SrcMask DstPrefix       DstMask
         Prot  DSCP  sport_opr sport_range    dport_opr dport_range     Grant
      1: 0.0.0.0         0       10.1.0.0        16
         17      ef  0         [1, 65535]      0        [1, 65535]      Permit
 Traffic Class Aggregate List:
   List: Prot  DSCP  sport_opr sport_range    dport_opr dport_range     Grant
      1: 17      ef  0         [1, 65535]      7        [3000, 4000]    Permit
 Keys:  protocol dscp DstPort
```

Step 3 **show ip cache verbose flow**

This is a NetFlow command that is used to display all the flows (including applications) currently active on the border router. The following example displays traffic flow statistics by protocol, source address, and destination:

Example:

```
Router# show ip cache verbose flow
IP packet size distribution (203337 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .397 .602 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000
    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000
IP Flow Switching Cache, 278544 bytes
  5 active, 4091 inactive, 310 added
  47486 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
IP Sub Flow Cache, 25800 bytes
  13 active, 1011 inactive, 355 added, 310 added to flow
  0 alloc failures, 0 force free
  1 chunk, 1 chunk added
  last clearing of statistics never
Protocol         Total    Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------         Flows     /Sec     /Flow  /Pkt     /Sec     /Flow     /Flow
TCP-other           14      0.0      7370    40      9.7    1556.8       3.4
UDP-other            9      0.0      7579    28      6.4    1601.0       3.5
ICMP               282      0.0         1    64      0.0       0.0      15.6
Total:             305      0.0       562    35     16.3     118.7      14.7

SrcIf          SrcIPaddress    DstIf          DstIPaddress    Pr TOS Flgs  Pkts
Port Msk AS                    Port Msk AS    NextHop              B/Pk  Active
Et8/0          172.20.1.1      Et0/0          10.1.3.1        11 B8  10    6334
07D0 /0  0                     0DAC /0  0     10.40.40.2              28  1337.8
Et8/0          172.20.1.1      Et0/0          10.2.2.1        06 00  00    6338
07D0 /0  0                     0DAC /0  0     10.40.40.2              40  1338.6
Et8/0          172.20.1.1      Et0/0          10.1.3.1        06 00  00    6333
07D0 /0  0                     0DAC /0  0     10.40.40.2              40  1337.6
Et8/0          172.20.1.1      Et0/0          10.1.1.1        06 00  00    6334
07D0 /0  0                     1964 /0  0     10.40.40.2              40  1337.8
Et8/0          172.20.1.1      Et0/0          10.1.1.1        11 B8  10    6339
07D0 /0  0                     0E10 /0  0     10.40.40.2              28  1338.8
Total number of prefixes 2
```

Step 4 **show oer border passive cache** {**learned** | **prefix**} [**applications]**

This command is used to display real-time prefix information collected from the border router through NetFlow passive monitoring. Using the **learned** and **applications** keywords you can display information about learned applications. In the output you can see that only application traffic classes matching the traffic class keys, filter, and aggregation criteria set in the first three tasks under the Specifying the Flow Keys for Automatic Learning of Application Traffic Classes task are saved in the learn cache.

Example:

```
Router# show oer border passive cache learned applications
OER Learn Cache:
    State is enabled
    Measurement type: throughput, Duration: 2 min
    Aggregation type: prefix-length, Prefix length: 24
    4096 oer-flows per chunk,
    8 chunks allocated, 32 max chunks,
    5 allocated records, 32763 free records, 4588032 bytes allocated
Prefix          Mask     Pkts  B/Pk  Delay Samples   Active
Prot  Dscp  SrcPort          DstPort
Host1           Host2           Host3           Host4           Host5
dport1          dport2          dport3          dport4          dport5
10.1.3.0         /24     873     28      0       0      13.3
17      ef [1, 65535]       [3000, 4000]
10.1.3.1        0.0.0.0         0.0.0.0         0.0.0.0         0.0.0.0
3500             0                0               0               0
10.1.1.0         /24    7674     28      0       0      13.4
17      ef [1, 65535]       [3000, 4000]
10.1.1.1        0.0.0.0         0.0.0.0         0.0.0.0         0.0.0.0
3600             0                0               0               0
```

- What To Do Next, page 30

What To Do Next

More information about monitoring and measuring traffic flow information for applications is documented in the “Measuring the Traffic Class Performance and Link Utilization Using OER” module.

Manually Selecting Prefixes for OER Monitoring

Perform this task to manually select prefixes for monitoring. An IP prefix list is created to define the prefix or prefix range. The prefix list is then imported into the central policy database by configuring a match clause in an OER map. For details about using IP prefix lists with OER, see Prefix Traffic Class Configuration Using OER.

- OER Map Operation for the OER Profile Phase, page 30
- What to Do Next, page 32

OER Map Operation for the OER Profile Phase

An OER map may appear to be similar to a route map but there are significant differences. An OER map is configured to select an IP prefix list using a match clause. The OER map is configured with a sequence number like a route map, and the OER map with the lowest sequence number is evaluated first. The operation of an OER map differs from a route map at this point. There are two important distinctions:

- Only a single match clause may be configured for each sequence. An error message will be displayed on the console if you attempt to configure multiple match clauses for a single OER map sequence.
- An OER map is not configured with permit or deny statements. However, a permit or deny sequence can be configured for an IP traffic flow by configuring a permit or deny statement in an IP prefix list and then applying the prefix list to the OER map.

Note Match precedence priority is not supported in OER maps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip prefix-list** *list-name* [**seq** *seq-value*]{**deny** *network/length* | **permit** *network/length*}[**le** *le-value*]
4. **oer-map map-name** *sequence-number*
5. **match ip address prefix-list** *name*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip prefix-list** *list-name* [**seq** *seq-value*]{**deny** *network/length* | **permit** *network/length*}[**le** *le-value*] **Example:** `Router(config)# ip prefix-list PREFIXES seq 20 permit 10.1.5.0/24`	Creates a prefix list to manually select prefixes for monitoring. • A master controller can monitor and control an exact prefix of any length including the default route. The master controller acts only on the configured prefix. • A master controller can monitor and control an inclusive prefix using the le 32 option. The master controller acts on the configured prefix and forces any more specific prefixes in the RIB to use the same exit. **Note** This option should be applied carefully. It is not needed in typical deployments. • The example creates an IP prefix list for OER to monitor and control the exact prefix, 10.1.5.0/24
Step 4	**oer-map map-name** *sequence-number* **Example:** `Router(config)# oer-map IMPORT 10`	Enters OER map configuration mode to create or configure an OER map. • *Only a single match clause can be configured for each OER map sequence.* • The example creates an OER map named IMPORT.

	Command or Action	Purpose
Step 5	**match ip address prefix-list** *name* **Example:** `Router(config-oer-map)# match ip address prefix-list PREFIXES`	Creates a prefix list match clause entry in an OER map to apply OER policies. • This command supports IP prefix lists only. • The example configures the prefix list PREFIXES.
Step 6	**end** **Example:** `Router(config-oer-map)# end`	Exits OER map configuration mode and returns to privileged EXEC mode.

What to Do Next

This section shows how to manually configure prefix learning. To configure automatic prefix learning, see the Configuring OER to Automatically Learn Prefix-Based Traffic Classes task.

Manually Selecting Inside Prefixes for OER Monitoring

In Cisco IOS Release 12.4(9)T, 12.2(33)SRB, and later releases, the OER BGP inbound optimization feature introduced the ability to manually select inside prefixes to support best entrance selection for traffic that originates from prefixes outside an autonomous system destined for prefixes inside the autonomous system. Perform this task to manually select inside prefixes for OER monitoring by creating an IP prefix list to define the inside prefix or prefix range. The prefix list is then imported into the MTC list by configuring a match clause in an OER map. For details about using IP prefix lists with OER, see Prefix Traffic Class Configuration Using OER .

- OER Inside Prefixes, page 32
- OER Map Operation for Inside Prefixes, page 32
- What to Do Next, page 34

OER Inside Prefixes

An OER inside prefix is defined as a public IP prefix assigned to a company. An OER outside prefix is defined as a public IP prefix assigned outside the company. Companies advertise the inside prefixes over the Internet using an Internet service provider (ISP) and receive advertisements for outside prefixes from an ISP.

OER Map Operation for Inside Prefixes

The operation of an OER map is similar to the operation of a route-map. An OER map is configured to select an IP prefix list or OER learn policy using a match clause and then to apply OER policy configurations using a set clause. The OER map is configured with a sequence number like a route-map, and the OER map with the lowest sequence number is evaluated first. In Cisco IOS Release 12.4(9)T and 12.2(33)SRB, the **inside** keyword that identifies inside prefixes was added to the **match ip address** (OER) command.

Note Match precedence priority is not supported in OER maps.

This task requires the master controller and border routers to be running Cisco IOS Release 12.4(9)T, 12.2(33)SRB, or later releases.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip prefix-list** *list-name* [**seq** *seq-value*]{**deny** *network/length* | **permit** *network/length*}[**le** *le-value*]
4. **oer-map map-name** *sequence-number*
5. **match ip address prefix-list** *name* [**inside**]
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip prefix-list** *list-name* [**seq** *seq-value*] {**deny** *network/length* | **permit** *network/length*}[**le** *le-value*] **Example:** `Router(config)# ip prefix-list INSIDE_PREFIXES seq 20 permit 192.168.1.0/24`	Creates a prefix list to manually select prefixes for monitoring. • A master controller can monitor and control an exact prefix of any length including the default route. The master controller acts only on the configured prefix. • A master controller can monitor and control an inclusive prefix using the le 32 option. The master controller acts on the configured prefix and forces any more specific prefixes in the RIB to use the same exit. **Note** This option should be applied carefully. It is not needed in typical deployments. • The example creates an IP prefix list for OER to monitor and control the exact prefix, 192.168.1.0/24

	Command or Action	Purpose
Step 4	**oer-map map-name** *sequence-number* **Example:** `Router(config)# oer-map INSIDE_MAP 10`	Enters OER map configuration mode to create or configure an OER map. • *OER map operation is similar to that of route maps.* • *Only a single match clause can be configured for each OER map sequence.* • Common and deny sequences should be applied to lowest OER map sequence for best performance. • The example creates an OER map named INSIDE_MAP.
Step 5	**match ip address prefix-list** *name* [**inside**] **Example:** `Router(config-oer-map)# match ip address prefix-list INSIDE_PREFIXES inside`	Creates a prefix list match clause entry in an OER map to apply OER policies. • This command supports IP prefix lists only. • Use the **inside** keyword to identify inside prefixes. • The example creates a match clause to use the prefix list INSIDE_PREFIXES to specify that inside prefixes must be matched.
Step 6	**end** **Example:** `Router(config-oer-map)# end`	Exits OER map configuration mode and returns to privileged EXEC mode.

What to Do Next

This section shows how to configure specific inside prefixes for OER monitoring and optimization. To configure automatic prefix learning for inside prefixes, see the Configuring OER to Automatically Learn Traffic Classes Using Inside Prefixes task.

Manually Selecting Traffic Classes Using Prefix Protocol Port and DSCP Value

Perform this task to manually select traffic classes using prefixes, protocols, port numbers, and DSCP value for OER monitoring. An IP access list is created to define the parameters to identify the traffic classes. The access list can then be imported into the MTC list by configuring a match clause in an OER map.

This example task uses an access list to identify voice traffic. Before voice traffic can be optimized, it must be identified. In this task, the voice traffic that is to be optimized is identified by a protocol of UDP, a range of source and destination port numbers from 16384 to 32767, a destination prefix of 10.20.20.0/24, and a DSCP value of ef.

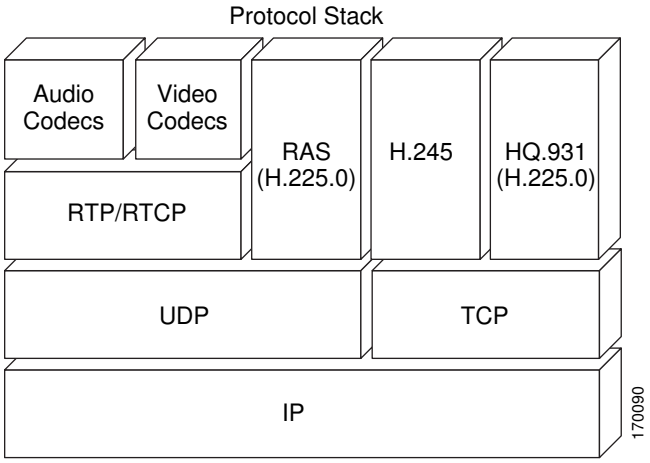
- IP Protocol Stack for Voice, page 34
- What to Do Next, page 37

IP Protocol Stack for Voice

Voice traffic uses a variety of protocols and streams on the underlying IP network. The figure below is a representation of the protocol options available for carrying voice traffic over IP. Most signaling traffic for

voice is carried over TCP. Most voice calls are carried over User Datagram Protocol (UDP) and Real-Time Protocol (RTP). You can configure your voice devices to use a specific range of destination port numbers over UDP to carry voice call traffic.

Figure 3 Protocol Stack Options Available for Voice Traffic



This task requires the master controller and border routers to be running Cisco IOS Release 12.4(9)T, 12.2(33)SRB, or later releases.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list** {**standard** | **extended**} *access-list-name*
4. [*sequence-number*] **permit udp** *source source-wildcard* [*operator* [*port*]] *destination destination-wildcard* [*operator* [*port*]] [**dscp** *dscp-value*]
5. **exit**
6. **oer-map** *map-name sequence-number*
7. **match ip address** {**access-list** *access-list-name*| **prefix-list** *prefix-list-name*}
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip access-list** {**standard** | **extended**} *access-list-name* **Example:** `Router(config)# ip access-list extended VOICE_ACCESS_LIST`	Defines an IP access list by name. • OER supports only named access lists. • The example creates an extended IP access list named VOICE_ACCESS_LIST.
Step 4	[*sequence-number*] **permit udp** *source source-wildcard* [*operator* [*port*]] *destination destination-wildcard* [*operator* [*port*]] [**dscp** *dscp-value*] **Example:** `Router(config-ext-nacl)# permit udp any range 16384 32767 10.20.20.0 0.0.0.15 range 16384 32767 dscp ef`	Sets conditions to allow a packet to pass a named IP access list. • The example is configured to identify all UDP traffic with a source or destination port number in the range from 16384 to 32767 from any source prefix to a destination prefix of 10.20.20.0/24, and with a DSCP value of ef. This specific UDP traffic represents voice traffic. • Only the syntax applicable to this task is shown. For more details, see the *Cisco IOS IP Application Services Command Reference*, Release 12.4T
Step 5	**exit** **Example:** `Router(config-ext-nacl)# exit`	(Optional) Exits extended access list configuration mode and returns to global configuration mode.
Step 6	**oer-map** *map-name sequence-number* **Example:** `Router(config)# oer-map VOICE_MAP 10`	Enters OER map configuration mode to configure an OER map to apply policies to selected IP prefixes. • *Only one match clause can be configured for each OER map sequence.* • Permit sequences are first defined in an IP prefix list and then applied with the **match ip address** (OER) command in Step 7. • The example creates an OER map named VOICE_MAP.
Step 7	**match ip address** {**access-list** *access-list-name*| **prefix-list** *prefix-list-name*} **Example:** `Router(config-oer-map)# match ip address access-list VOICE_ACCESS_LIST`	References an extended IP access list or IP prefix as match criteria in an OER map. • Only a single match clause can be configured for each OER map sequence. • The example configures the IP access list named VOICE_ACCESS_LIST as match criteria in an OER map.

	Command or Action	Purpose
Step 8	**end** **Example:** `Router(config-oer-map)# end`	(Optional) Exits OER map configuration mode and returns to privileged EXEC mode.

What to Do Next

This section shows how to manually select traffic classes using prefixes, protocols, port numbers, and DSCP value for OER monitoring. To configure automatic learning of traffic classes using prefixes, protocols, port numbers, and DSCP values, see the Specifying the Flow Keys for Automatic Learning of Application Traffic Classes task.

Configuration Examples for Using OER to Profile the Traffic Classes

- Configuring OER to Automatically Learn Prefix-Based Traffic Classes Example, page 37
- Configuring OER to Automatically Learn Traffic Classes Using Inside Prefixes Example, page 38
- ConfiguringOERtoAutomaticallyLearnTrafficClassesUsingPrefixesandProtocolorPortNumbers Example, page 38
- Configuring OER to Automatically Learn Traffic Classes Using Protocol Ports and DSCP Value Example, page 38
- Manually Selecting Prefixes for OER Monitoring Example, page 39
- Manually Selecting Inside Prefixes for OER Monitoring Example, page 40
- ManuallySelectingTrafficClassesUsingPrefix Protocol Port andDSCPValue Example, page 40

Configuring OER to Automatically Learn Prefix-Based Traffic Classes Example

The following example, starting in global configuration mode, configures the master controller to automatically learn top prefixes based on the highest delay. The prefix monitoring period is set to 10 minutes. The number of prefixes that are monitored during each monitoring period is set to 500. The time interval between each monitoring period is set to 20 minutes.

```
Router(config)# oer master
Router(config-oer-master)# learn
Router(config-oer-master-learn)# delay
Router(config-oer-master-learn)# aggregation-type bgp

Router(config-oer-master-learn)# monitor-period 10

Router(config-oer-master-learn)# periodic-interval 20

Router(config-oer-master-learn)# prefixes 500
Router(config-oer-master-learn)# end
```

Configuring OER to Automatically Learn Traffic Classes Using Inside Prefixes Example

The following example shows how to configure OER to automatically learn prefixes inside the network:

```
Router> enable
Router#
configure terminal
Router(config)# oer master
Router(config-oer-mc)# learn
Router(config-oer-mc-learn)# inside bgp
Router(config-oer-mc-learn)# monitor-period 10
Router(config-oer-mc-learn)# periodic-interval 20

Router(config-oer-mc-learn)# prefixes 500

Router(config-oer-mc-learn)# end
```

ConfiguringOERtoAutomaticallyLearnTrafficClassesUsingPrefixesandProtocolorPortNumbers Example

The following example, starting in global configuration mode, learns traffic for SSH sessions that use 49152 as the destination port number in the IP packet header.

```
Router(config)# oer master

Router(config-oer-master)# learn
Router(config-oer-master-learn)# throughput
Router(config-oer-master-learn)# aggregation-type bgp

Router(config-oer-master-learn)# monitor-period 10

Router(config-oer-master-learn)# periodic-interval 20

Router(config-oer-master-learn)# protocol 22 port 49152 dst
Router(config-oer-master-learn)# end
```

Configuring OER to Automatically Learn Traffic Classes Using Protocol Ports and DSCP Value Example

The following example, starting in global configuration mode, configures the master controller to automatically learn defined application traffic. Using a series of traffic class commands under OER learn configuration mode, only voice traffic with a DSCP bit set to ef, a protocol of UDP, and a destination port in the range of 3000 to 4000 is learned and added to the OER MTC list on the master controller.

The prefix monitoring period is set to 2 minutes. The number of prefixes that are monitored during each monitoring period is set to 10. The time interval between each monitoring period is set to 20 minutes.

```
Router(config)# ip access-list extended voice-filter-acl
Router(config-ext-nacl)# permit udp any 10.1.0.0 0.0.255.255 dscp ef

Router(config-ext-nacl)# exit
Router(config)# ip access-list extended voice-agg-acl
Router(config-ext-nacl)# permit udp any any range 3000 4000 dscp ef

Router(config-ext-nacl)# exit
Router(config)# oer master
```

```
Router(config-oer-master)# learn

Router(config-oer-master-learn)# aggregation-type prefix-length 24

Router(config-oer-master-learn)# throughput

Router(config-oer-master-learn)# monitor-period 2
Router(config-oer-master-learn)# periodic-interval 1

Router(config-oer-master-learn)# prefixes 10

Router(config-oer-master-learn)# traffic-class filter access-list voice-filter-acl

Router(config-oer-master-learn)# traffic-class aggregate access-list voice-agg-acl

Router(config-oer-master-learn)# traffic-class keys protocol dport dscp
Router(config-oer-master-learn)# end
```

More details about the OER network configuration for the example shown above can be seen in the running configuration file:

```
Router# show running-config
oer master
port 7777
logging
!
border 10.1.1.1 key-chain key1
 interface Serial12/0 external
 interface Ethernet8/0 internal
!
border 10.1.1.2 key-chain key2
 interface Ethernet0/0 external
 interface Ethernet8/0 internal
!
learn
 throughput
 periodic-interval 1
 monitor-period 2
 prefixes 10
 traffic-class filter access-list voice-filter-acl
 traffic-class aggregate access-list voice-agg-acl
 traffic-class keys protocol dscp dport
 backoff 180 200
 mode route control
 mode monitor active
!
active-probe echo 10.1.2.1
active-probe echo 10.1.1.1
active-probe echo 10.1.3.1
```

Manually Selecting Prefixes for OER Monitoring Example

The following example, starting in global configuration mode, configures an OER map to exclude traffic from the 192.168.0.0/16 network and include traffic from the 10.5.5.0/24 network. Excluded prefixes are not imported into the MTC list.

```
Router(config)# ip prefix-list seq 10 EXCLUDE deny 192.168.0.0/16 le 32

Router(config)# ip prefix-list seq 10 IMPORT permit 10.5.5.0/24

Router(config)# oer-map PREFIXES 10
Router(config-oer-map)# match ip address prefix-list EXCLUDE
Router(config-oer-map)# exit

Router(config)# oer-map PREFIXES 20

Router(config-oer-map)# match ip address prefix-list IMPORT

Router(config-oer-map)# end
```

Manually Selecting Inside Prefixes for OER Monitoring Example

The following example shows how to manually configure OER to learn prefixes inside the network using an OER map:

```
Router> enable
Router# configure terminal
Router(config)# ip prefix-list INSIDE_PREFIXES seq 20 permit 192.168.1.0/24
Router(config)# oer-map INSIDE_MAP 10
Router(config-oer-map)# match ip address prefix-list INSIDE_PREFIXES inside
Router(config-oer-map)# end
```

ManuallySelectingTrafficClassesUsingPrefix Protocol Port andDSCPValue Example

The following configuration is performed on an edge router which is both an OER master controller and a border router (for example, in a remote office network) to identify voice traffic using an extended named access list.

```
Router> enable
Router# configure terminal
Router(config)# ip access-list extended Voice_Traffic
Router(config-ext-nacl)# 10 permit udp any 10.1.0.0 0.0.255.255 range 16384 32767 dscp ef
Router(config-ext-nacl)# exit
Router(config)# oer-map Voice_MAP 10
Router(config-oer-map)# match ip address access-list Voice_Traffic
Router(config-oer-map)# end
```

Where To Go Next

This module covered the OER profile phase and it has assumed that you started with the Cisco IOS Optimized Edge Routing Overview and the Setting Up OER Network Components modules. The profile phase is the first phase in the OER performance loop. To learn more about the other OER phases, read through the other modules in the following list:

- Measuring the Traffic Class Performance and Link Utilization Using OER
- Configuring and Applying OER Policies
- Using OER to Control Traffic Classes and Verify the Route Control Changes

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS Master Command List	http://www.cisco.com/en/US/docs/ios/mcl/allreleasemcl/all_book.html
Command Lookup Tool	http://tools.cisco.com/Support/CLILookup

Related Topic	Document Title
Cisco OER technology overview	Cisco IOS Optimized Edge Routing Overview module
Concepts and configuration tasks required to set up OER network components.	Setting Up OER Network Components module
Cisco OER commands: complete command syntax, command mode, command history, defaults, usage guidelines and examples	*Cisco IOS Optimized Edge Routing Command Reference*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Using OER to Profile the Traffic Classes

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1 Feature Information for Using OER to Profile the Traffic Classes

Feature Name	Releases	Feature Configuration Information
Port and Protocol Based Prefix Learning	12.3(11)T 12.2(33)SRB	Port and protocol based prefix learning allows you to configure a master controller to learn prefixes based on the protocol type and TCP or UDP port number. The **protocol** command was introduced by this feature.

Feature Name	Releases	Feature Configuration Information
expire command[1]	12.3(14)T 12.2(33)SRB	The **expire** command is used to set an expiration period for learned prefixes. By default, the master controller removes inactive prefixes from the central policy database as memory is needed. This command allows you to refine this behavior by setting a time or session based limit. The time based limit is configured in minutes. The session based limit is configured for the number of monitor periods (or sessions).
OER Application-Aware Routing: PBR	12.4(2)T 12.2(33)SRB	The OER Application-Aware Routing: PBR feature introduces the capability to optimize IP traffic based on the type of application that is carried by the monitored prefix. Independent policy configuration is applied to the subset (application) of traffic. The following commands were introduced or modified by this feature: **debug oer border pbr**, **debug oer master prefix**, **match ip address (OER)**, **show oer master active-probes**, and **show oer master appl**.

1 This is a minor enhancement. Minor enhancements are not typically listed in Feature Navigator.

Feature Name	Releases	Feature Configuration Information
OER BGP Inbound Optimization	12.4(9)T 12.2(33)SRB	OER BGP inbound optimization supports best entrance selection for traffic that originates from prefixes outside an autonomous system destined for prefixes inside the autonomous system. External BGP (eBGP) advertisements from an autonomous system to an Internet service provider (ISP) can influence the entrance path for traffic entering the network. OER uses eBGP advertisements to manipulate the best entrance selection. The following commands were introduced or modified by this feature: **clear oer master prefix**, **downgrade bgp**, **inside bgp**, **match ip address (OER)**, **match oer learn**, **max range receive**, **maximum utilization receive**, **show oer master prefix**.

Feature Name	Releases	Feature Configuration Information
OER DSCP Monitoring	12.4(9)T 12.2(33)SRB	OER DSCP Monitoring introduced automatic learning of traffic classes based on protocol, port numbers, and DSCP value. Traffic classes can be defined by a combination of keys comprising of protocol, port numbers, and DSCP values, with the ability to filter out traffic that is not required, and the ability to aggregate the traffic in which you are interested. Layer 4 information such as protocol, port number, and DSCP information is now sent to the master controller database in addition to the Layer 3 prefix information. The new functionality allows OER to both actively and passively monitor application traffic. The following commands were introduced or modified by this feature: **show oer border passive applications**, **show oer border passive cache**, **show oer border passive learn**, **show oer master appl**, **traffic-class aggregation**, **traffic-class filter**, and **traffic-class keys**.

Feature Name	Releases	Feature Configuration Information
OER Border Router Only Functionality	12.2(33)SXH	In Cisco IOS Release 12.2(33)SXH support for using a Cisco Catalyst 6500 series switch as an OER border router was introduced. Only border router functionality is included in the Cisco IOS Release 12.2(33)SXH images; no master controller configuration is available. The master controller that communicates with the Cisco Catalyst 6500 series switch being used as a border router must be a router running Cisco IOS Release 12.4(6)T or a later release. The OER master controller software has been modified to handle the limited functionality supported by the Cisco Catalyst 6500 border routers. Using the Route Processor (RP), the Catalyst 6500 border routers can capture throughput statistics only for a traffic class compared to the delay, loss, unreachability, and throughput statistics collected by non-Catalyst 6500 border routers. A master controller automatically detects the limited capabilities of the Catalyst 6500 border routers and downgrades other border routers to capture only the throughput statistics for traffic classes. By ignoring other types of statistics, the master controller is presented with a uniform view of the border router functionality. The following command was introduced or modified by this feature: **show oer border passive cache.**

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

© 2011 Cisco Systems, Inc. All rights reserved.