



Cable ARP Filtering

First Published: February 14, 2008

Last Updated: February 9, 2009



Note

Cisco IOS Release 12.2(33)SCA integrates support for this feature on the Cisco CMTS routers. This feature is also supported in Cisco IOS Release 12.3BC, and this document contains information that references many legacy documents related to Cisco IOS 12.3BC. In general, any references to Cisco IOS Release 12.3BC also apply to Cisco IOS Release 12.2SC.

This document describes the Cable ARP Filtering feature for the Cisco Cable Modem Termination System (CMTS). This feature enables service providers to filter Address Resolution Protocol (ARP) request and reply packets, to prevent a large volume of such packets from interfering with the other traffic on the cable network.

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 <http://tools.cisco.com/ITDIT/CFN/>. An account on <http://www.cisco.com/> is not required.

Content

- [Prerequisites for Cable ARP Filtering, page 2](#)
- [Restrictions for Cable ARP Filtering, page 3](#)
- [Information About Cable ARP Filtering, page 3](#)
- [How to Configure Cable ARP Filtering, page 7](#)
- [Configuration Examples for Cable ARP Filtering, page 16](#)
- [Additional References, page 19](#)

- [Feature Information for Cable ARP Filtering on the Cisco Cable Modem Termination System, page 20](#)

Prerequisites for Cable ARP Filtering

The Cable ARP Filtering feature is supported on the Cisco CMTS routers in Cisco IOS software release trains 12.3BC and 12.2SC. [Table 1: Cable ARP Filtering Hardware Compatibility Matrix, on page 2](#) shows the hardware compatibility prerequisites for this.

Table 1: Cable ARP Filtering Hardware Compatibility Matrix

CMTS Platform	Processor Engine	Cable Interface Cards
Cisco uBR10012 Universal Broadband Router	<p>Cisco IOS Release 12.2(33)SCA and later</p> <ul style="list-style-type: none"> • PRE2 <p>Cisco IOS Release 12.2(33)SCB and later</p> <ul style="list-style-type: none"> • PRE4 <p>Cisco IOS Release 12.2(33)SCH and later</p> <ul style="list-style-type: none"> • PRE5 	<p>Cisco IOS Release 12.2(33)SCA and later</p> <ul style="list-style-type: none"> • Cisco uBR10-MC5X20S/U¹ <p>Cisco IOS Release 12.2(33)SCC and later</p> <ul style="list-style-type: none"> • Cisco UBR-MC20X20V² <p>Cisco IOS Release 12.2(33)SCE and later</p> <ul style="list-style-type: none"> • Cisco uBR-MC3GX60V 2
Cisco uBR7246VXR Universal Broadband Router	<p>Cisco IOS Release 12.2(33)SCA</p> <ul style="list-style-type: none"> • NPE-G1 <p>Cisco IOS Release 12.2(33)SCB</p> <ul style="list-style-type: none"> • NPE-G2 	<p>Cisco IOS Release 12.2(33)SCA</p> <ul style="list-style-type: none"> • Cisco uBR-MC28U/X • Cisco uBR-MC16U/X <p>Cisco IOS Release 12.2(33)SCD and later</p> <ul style="list-style-type: none"> • Cisco uBR-MC88V 2
Cisco uBR7225VXR Universal Broadband Router	<p>Cisco IOS Release 12.2(33)SCA</p> <ul style="list-style-type: none"> • NPE-G1 <p>Cisco IOS Release 12.2(33)SCB</p> <ul style="list-style-type: none"> • NPE-G2 	<p>Cisco IOS Release 12.2(33)SCA</p> <ul style="list-style-type: none"> • Cisco uBR-E-28U • Cisco uBR-E-16U • Cisco uBR-MC28U/X • Cisco uBR-MC16U/X <p>Cisco IOS Release 12.2(33)SCD and later</p> <ul style="list-style-type: none"> • Cisco uBR-MC88V 2

- 1 Supports only DOCSIS 2.0 and IPv6 cable modems.
- 2 Supports only DOCSIS 3.0 and IPv6 cable modems.

Restrictions for Cable ARP Filtering

Cisco uBR7100 Series Restrictions

- The Cable ARP Filtering feature is not supported on the Cisco uBR7100 series universal broadband routers.

Cisco uBR10012 Router Restrictions

- The Cisco uBR10012 router maintains ARP filtering statistics on the Performance Routing Engine (PRE) module. Statistics are viewed with the **show cable arp-filter** command for a specified interface. When a switchover event occurs, as in RPR+ Redundancy, these ARP filtering statistics are reset to zero.

Cisco uBR10012 PRE modules support the Route Processor Redundancy Plus (RPR+) feature in several Cisco IOS releases. Refer to the following document for additional information:

Route Processor Redundancy Plus for the Cisco uBR10012 Universal Broadband Router

<http://www.cisco.com/en/US/docs/cable/cmts/feature/u10krprp.html>

- The Cable ARP Filter feature is not configurable per subinterface.

PXF ARP Filter Restrictions

- The PXF microcode must be enhanced to provide the rate limiting functionality for ARP filtering in PXF.
- ARP filtering in PXF is only supported on the Performance Routing Engine 2 (PRE2) and later versions. For more information, refer to the [ARP Filtering in PXF, on page 5](#).
- The ARP Filter in PXF feature is not configurable per subinterface.

Information About Cable ARP Filtering

Overview

Theft-of-service and denial-of-service (DNS) attacks have become increasingly common in cable broadband networks. In addition, virus attacks are becoming more common, and users are often unaware that their computers have become infected and are being used to continue the attacks on the network.

One sign that often appears during these attacks is an unusually high volume of Address Resolution Protocol (ARP) packets. The user or virus repeatedly issues ARP requests, trying to find the IP addresses of additional computers that might be vulnerable to attack.

ARP requests are broadcast packets, so they are broadcast to all devices on that particular network segment. In some cases, a router can also forward ARP broadcasts to an ARP proxy for further processing.

This problem is also made worse because some low-end routers commonly used by subscribers for home networks can also incorrectly respond to all ARP requests, which generates even more traffic. Until these customer premises equipment (CPE) devices can be upgraded with firmware that is compliant to the appropriate Request for Comments (RFC) specifications, service providers need to be able to deal with the incorrectly generated or forwarded traffic.

In addition, the Cisco CMTS router automatically monitors ARP traffic and enters the IP addresses found in ARP requests into its own ARP table, in the expectation that a device will eventually be found with that IP address. Unacknowledged IP addresses remain in the router's ARP table for 60 seconds, which means that a large volume of ARP traffic can fill the router's ARP table.

This process can create a large volume of ARP traffic across the network. In some situations, the volume of ARP requests and replies can become so great that it can throttle other traffic and occupy most of the Cisco CMTS router's processing time, hampering efforts by technicians to recover their network.

The router cannot use fast-switching to process ARP packets, but must instead forward them to the route processor (RP). Because of this, processing a large volume of ARP traffic can also prevent the router from handling normal traffic.

Filtering ARP Traffic

To control the volume of ARP traffic on a cable interface, you can configure the **cable arp filter** command to specify how many ARP packets are allowed per Service ID (SID) during a user-specified time period. You can configure separate thresholds for ARP request packets and for ARP reply packets.

When a cable interface is configured to filter ARP packets, it maintains a table of the number of ARP request or reply packets that have been received for each SID. If a SID exceeds the maximum number of packets during the window time period, the Cisco CMTS drops the packets until a new time period begins.



Note

If using bundled cable interfaces, the Cable ARP Filtering feature is configured on the master and slave interfaces separately. This allows you to configure the feature only on the particular interfaces that require it. In addition, you can configure the feature with different threshold values, allowing you to customize the feature for each interface's traffic patterns.

Monitoring Filtered ARP Traffic

After ARP filtering has been enabled on a cable interface, you can then use the service **divert-rate-limit** command to display the devices that are generating excessive amounts of ARP traffic. These devices could be generating this traffic for any of the following reasons:

- Cable modems that are running software images that are either not DOCSIS-compliant or that have been hacked to allow theft-of-service attacks.
- CPE devices that are either performing a theft-of-service or denial-of-service attack, or that have been infected with a virus that is searching for other computers that can be infected.
- Routers or other devices that mistakenly reply to or forward all ARP requests.

After identifying the specific devices that are generating this traffic, you can use whatever techniques are allowed by your service level agreements (SLAs) to correct the problem.

Linksys Wireless-Broadband Router (BEFW11S4)

The Linksys Wireless-B Broadband Router, Model number BEFW11S4 version 4 with 1.44.2 firmware, incorrectly sends its own ARP reply packet for every ARP request packet it receives, instead of replying only to the ARP requests that are specifically for itself. Customers with these routers should upgrade the firmware to the latest revision to fix this bug. To upgrade the firmware, go to the download section on the Linksys website.

**Note**

It is extremely important that non-compliant CPE devices be updated to firmware that correctly handles ARP and other broadcast traffic. Even one or two non-compliant devices on a segment can create a significant problem with dropped packets, impacting all of the other customers on that segment.

ARP Filtering in PXF

Cisco Release 12.3(17a)BC introduces a PXF component to the ARP filter feature. When enabled, this PXF component filters ARP packets for identified ARP offenders, decreasing the ARP punt rate and RP CPU usage. It also provides the user with clearer separation in ARP filtering by utilizing source MAC addresses instead of SIDs.

The filter logic now filters by source MAC address instead of by SID. Currently, the modem MAC addresses are excluded from having their ARPs filtered, but Multimedia Terminal Adapters (MTAs) and other non-offending CPEs can still (statistically) have ARPs filtered because all ARPs appear to come from the same SID. Therefore, filtering by source MAC address will isolate the filtering to the offensive devices. By doing so, a customer who has Voice-over-IP (VoIP) service via an MTA and an infected CPE will not have MTA issues while being contacted by the service provider in regards to the infected CPE.

ARP offenders will still be allowed to use ARP to avoid complete loss of Internet connectivity through their configured or provisioned gateway address. Because of this, it is expected that the “ARP Input” process will still show a few percentage points of CPU usage, but the net interrupt CPU usage will decrease.

**Note**

ARP filtering in PXF is only supported on the PRE2 and later versions, and is enabled by default.

Filtering ARP Traffic in PXF

When ARP traffic in PXF is enabled, a lightweight algorithm executing on the RP is used to identify ARP offenders by the source MAC address or the SID. All offending source MAC addresses or SIDs are then programmed by the ARP Filter control module into the PXF ucode divert rate limiting module (ARP offenders are still allowed to perform ARP transactions, but only at the configured filtering rate).

Offending source MAC addresses or SIDs are filtered in PXF for a minimum of 50 minutes (ten 5-minute intervals with no occurring offenses). Utilizing the existing ARP Filter CLI tools, the cable operator can obtain enough information about the modem and CPE to contact the end user to request the necessary anti-virus software installation or firmware upgrade for the CPE.

**Note**

If the offending device is not “repaired” or shut off, it will remain in the PXF ARP Filter indefinitely.

The PXF ARP rate limiter is designed to filter a maximum of 16,000 ARP offenders. If this pool of 16,000 filterable entities is exhausted, then the entity is filtered on the RP. The CLI statistics will distinguish mac addresses filtered on the RP verses PXF.

Because of possible mac address hash collisions, ARP offenders that cannot be programmed into the PXF ARP rate limiter will still be filtered in PXF by SID. Since the hash is done by source mac address and SID, such devices can actually moved back to mac address filtering by deleting the associated modem and forcing it back online with a new SID (this merely a possibility and is not expected to be a common practice).

ARP packets with a source mac address that is not “known” to the CMTS as a modem or CPE will be filtered by their SID in PXF. Therefore, there will never be an unusual ARP packet source that will NOT be filtered in PXF. False ARP packets with invalid operation codes will be filtered as if they are an ARP Reply.

**Note**

ARP filtering in PXF is only supported on the PRE2 and later versions.

PXF Divert-Rate-Limit

Diverted packets sent from the forwarding processor (FP) to the route processor (RP), via the FP-to-RP interface, may encounter congestion when packets requiring diversion arrive at the FP at a faster rate than they can be transmitted to the RP. When congestion occurs, valid packets in the FP-to-RP queues will be dropped. This situation can be deliberately caused by attacks directed at the CMTS or inadvertently by faulty external hardware.

PXF Divert-Rate-Limit identifies packet streams that will cause congestion of the FP-to-RP interface. Packets in the stream are dropped according to the configured rate-limiting parameters. Rate-limiting occurs before the packets are placed in the FP-to-RP queues, preventing valid packets in other streams from being dropped.

The following diverted packets will be rate-limited:

- fwd-glean—Packets that hit a glean adjacency in the Forwarding Information Base (FIB).
- rpf-glean—Packets that hit a glean adjacency during the Reverse Path Forwarding (RPF) check.

Packets that pass rate-limiting are diverted as they normally would be. Packets that fail rate-limiting are dropped.

Rate-limiting is implemented by a token-bucket algorithm. The token-bucket algorithm requires two variables: rate and limit. Both the rate and limit are configurable via the CLI. The rate is the average number of packets-per-second that pass the rate-limiting code. The limit can be thought of as the number of packets that will pass during an initial burst of packets.

**Note**

The Divert-Rate-Limit feature is always on and cannot be turned off. Using the no form of the configuration CLI returns the rate-limiting parameters to their default values. During a PXF and CPU switchover or reload, the configuration is retained, but not the statistics. Therefore, after switchover, the statistics shown by the show pxf cpu statistics drl command will show zero.

fwd-glean

IP packets that hit a glean adjacency in the FIB are diverted. There are three requirements:

- RPF-check has passed (if required).
- SV-check has passed (if required).
- Forward adjacency is glean.

Packets are rate-limited based on the destination IP address. A hash on the destination IP address is used to create an index that stores state information for rate-limiting. In the event of a hash collision, the pre-existing state information will be used and updated. The table that stores state information is large enough to make collisions rare.

rpf-glean

The RPF feature is modified to divert packets that hit a glean adjacency during the RPF check. A new divert_code will be created for this type of diverted packet. Currently, these packets are dropped.

There are four requirements:

- SV-check has passed (if required).
- RPF is enabled.
- The packet is from a non-load-balanced interface.
- RPF-adjacency is glean.

Packets are rate-limited based on the source IP address. A hash on the source IP address is used to create an index that stores state information for rate-limiting. In the event of a hash collision, the pre-existing state information will be updated. The table that stores state information is large enough to make collisions rare.

How to Configure Cable ARP Filtering

Use the following procedures to determine whether ARP filtering is required and to configure ARP filtering on one or more cable interfaces.

Monitoring ARP Processing

Use the following steps to monitor how the router is processing ARP traffic and whether the volume of ARP packets is a potential problem.

Step 1

To discover the CPU processes that are running most often, use the **show process cpu sorted** command and look for the ARP Input process:

Example:

```
Router# show process cpu sorted
```

```

CPU utilization for five seconds: 99%/28%; one minute: 93%; five minutes: 90%
PID Runtime(ms)   Invoked    uSecs   5Sec   1Min   5Min  TTY Process
 19   139857888     44879804   3116   31.44% 28.84% 28.47% 0 ARP Input
154   74300964     49856254   1490   20.29% 19.46% 15.78% 0 SNMP ENGINE
 91   70251936     1070352    65635   8.92%  9.62%  9.59% 0 CEF process
 56   17413012     97415887    178   3.01%  3.67%  3.28% 0 C10K BPE IP Enqu
 78   24985008     44343708    563   3.68%  3.47%  3.24% 0 IP Input
 54    6075792     6577800    923   0.90%  0.67%  0.65% 0 CMTS SID mgmt ta
...

```

In this example, the ARP Input process has used 31.44 percent of the CPU for the past five seconds. Total CPU utilization is also at 99 percent, indicating that a major problem exists on the router.

Note As a general rule, the ARP Input process should use no more than one percent of CPU processing time during normal operations. The ARP Input process could use more processing time during certain situations, such as when thousands of cable modems are registering at the same time, but if it uses more than one percent of processing time during normal operations, it probably indicates a problem.

Step 2 To monitor only the ARP processes, use the **show process cpu | include ARP** command:

Example:

```

Router# show process cpu | include ARP

 19   139857888   44879804       3116 31.44% 28.84% 28.47% 0 ARP Input
110           0           1           0 0.00% 0.00% 0.00% 0 RARP Input

```

Step 3 To monitor the number of ARP packets being processed, use the **show ip traffic** command.

Example:

```

Router# show ip traffic | begin ARP

ARP statistics:
  Rcvd: 11241074 requests, 390880354 replies, 0 reverse, 0 other
  Sent: 22075062 requests, 10047583 replies (2127731 proxy), 0 reverse

```

Repeat this command to see how rapidly the ARP traffic increases.

Step 4 If ARP traffic appears to be excessive, use the **show cable arp-filter** command to display ARP traffic for each cable interface, to identify the interfaces that are generating the majority of the traffic.

Example:

```

Router# show cable arp-filter Cable5/0/0

ARP Filter statistics for Cable5/0/0:
  Rcvd Replies: 177387 total, 0 unfiltered, 0 filtered
  Sent Requests For IP: 68625 total, 0 unfiltered, 0 filtered
  Sent Requests Proxied: 7969175 total, 0 unfiltered, 0 filtered

```

In the above example, the unfiltered and filtered counters show zero, which indicates that ARP filtering has not been enabled on the cable interface. After ARP filtering has been enabled with the **cable arp filter** command, you can identify the specific devices that are generating excessive ARP traffic by using the **service divert-rate-limit** command (see the [Identifying the Sources of Major ARP Traffic](#), on page 10).

Enabling ARP Filtering

Use the following procedure to enable ARP filtering on a particular cable interface.

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	interface cable x/y Example: Router(config)# interface cable 5/1	Enters interface configuration mode for the specified cable interface.
Step 4	cable arp filter reply-accept number window-size Example: Router(config-if)# cable arp filter reply-accept 2 2	Configures the cable interface to accept only the specified <i>number</i> of ARP reply packets every <i>window-size</i> seconds for each active Service ID (SID) on that interface. The cable interface drops ARP reply packets for a SID that would exceed this number. (The default behavior is to accept all ARP reply packets.)
Step 5	cable arp filter request-send number window-size Example: Router(config-if)# cable arp filter request-send 3 1	Configures the cable interface to send only the specified <i>number</i> of ARP request packets every <i>window-size</i> seconds for each active SID on that interface. The cable interface drops ARP requests for a SID that would exceed this number. (The default behavior is to send all ARP request packets.) Note Repeat Step 3 through Step 5 to enable ARP filtering on other cable interfaces. Master and slave interfaces in a cable bundle must be configured separately.
Step 6	end Example: Router(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.

Identifying the Sources of Major ARP Traffic

After you have begun filtering ARP traffic on a cable interface, use the following procedure to identify the cable modems or CPE devices that are generating or forwarding major amounts of ARP traffic.



Tip

The Linksys Wireless-B Broadband Router, Model number BEFW11S4 version 4 with 1.44.2 firmware, has a known problem in which it incorrectly generates an ARP reply for every ARP request packet it receives. See the [Linksys Wireless-Broadband Router \(BEFW11S4\)](#) guide for information on how to resolve this problem.

Step 1 To discover the devices that are responsible for generating or forwarding more ARP requests on a specific cable interface than a specified minimum number of packets, use the **show cable arp-filter requests-filtered** command where *number* is the threshold value for the number of packets being generated:

Example:

```
show cable arp-filter cable interface requests-filtered number
```

For example, to display the devices that have generated more than 100 ARP request packets, enter the following command:

Example:

```
Router# show cable arp-filter cable 5/1/0 requests-filtered 100
```

Sid	MAC Address	IP Address	Req-Filtered	Req-For-IP-Filtered	Rep-Filtered
1	0006.2854.72d7	10.3.81.4	12407	0	0
81	00C0.c726.6b14	10.3.81.31	743	0	0

Step 2 Repeat the **show cable arp-filter** command to show how quickly the devices are generating the ARP packets.

Step 3 To discover the devices that are responsible for generating or forwarding more ARP replies on a specific cable interface than a specified minimum number of packets, use the **show cable arp-filter replies-filtered** command where *number* is the threshold value for the number of packets being generated:

Example:

```
show cable arp-filter cable interface requests-filtered number
```

For example, to display the devices that have generated more than 200 ARP reply packets, enter the following command:

Example:

```
Router# show cable arp-filter cable 5/0/0 replies-filtered 200
```

Sid	MAC Address	IP Address	Req-Filtered	Req-For-IP-Filtered	Rep-Filtered
2	0006.53b6.562f	10.11.81.16	0	0	2358
191	0100.f31c.990a	10.11.81.6	0	0	11290

Step 4 (Optional) If a particular cable modem is generating or forwarding excessive ARP replies, contact the customer to see if they are using a Linksys Wireless-B Broadband Router, Model number BEFW11S4. If so, this router could be running

old firmware that is incorrectly generating excessive ARP packets, and the customer should upgrade their firmware. For more information, see the [Linksys Wireless-Broadband Router \(BEFW11S4\)](#) guide

Step 5 Repeat this command during each filter period (the time period you entered with the **cable arp filter** command) to show how quickly the devices are generating the ARP packets.

Step 6 (Optional) The ARP reply and request packet counters are 16-bit counters, so if a very large number of packets are being generated on an interface, these counters could wrap around to zero in a few hours or even a few minutes. Clearing the ARP counters eliminates stale information from the display and makes it easier to see the worst offenders when you suspect ARP traffic is currently creating a problem on the network.

To eliminate the modems that are not currently triggering the ARP filters and to isolate the worst current offenders, use the **clear counters cable interface** command to reset all of the interface counters to zero. Then the **show cable arp-filter** commands clearly identify the SIDs of the modems that are currently forwarding the most ARP traffic.

For example, the following example indicates that a number of modems are forwarding a large enough volume of ARP traffic that they have triggered the ARP packet filters:

Example:

```
Router# show cable arp-filter cable 5/1/0 requests-filtered 10
```

Sid	MAC Address	IP Address	Req-Filtered	Req-For-IP-Filtered	Rep-Filtered
1	0006.2854.72d7	10.3.81.4	8	0	0
23	0007.0e02.b747	10.3.81.31	32	0	0
57	0007.0e03.2c51	10.3.81.31	12407	0	0
...					
81	00C0.c726.6b14	10.3.81.31	23	0	0

SID 57 shows the largest number of packets, but it is not immediately apparent if this modem is causing the current problems. After clearing the counters though, the worst offenders are easily seen:

Example:

```
Router# clear counter cable 5/1/0
```

```
Clear show interface counters on this interface [confirm] y
```

```
08:17:53.968: %CLEAR-5-COUNTERS: Clear counter on interface Cable5/1/0 by console
```

```
Router# show cable arp cable 5/1/0
```

```
ARP Filter statistics for Cable3/0:
  Replies Rcvd: 0 total. 0 unfiltered, 0 filtered
  Requests Sent For IP: 0 total. 0 unfiltered, 0 filtered
  Requests Forwarded: 0 total. 0 unfiltered, 0 filtered
```

```
Router# show cable arp-filter cable 5/1/0 requests-filtered 10
```

Sid	MAC Address	IP Address	Req-Filtered	Req-For-IP-Filtered	Rep-Filtered
Router# show cable arp-filter cable 5/1/0 requests-filtered 10					

Sid	MAC Address	IP Address	Req-Filtered	Req-For-IP-Filtered	Rep-Filtered
57	0007.0e03.2c51	10.3.81.31	20	0	0
81	00C0.c726.6b14	10.3.81.31	12	0	0

```
Router# show cable arp-filter cable 5/1/0 requests-filtered 10
```

Sid	MAC Address	IP Address	Req-Filtered	Req-For-IP-Filtered	Rep-Filtered
57	0007.0e03.2c51	10.3.81.31	31	0	0
81	00C0.c726.6b14	10.3.81.31	18	0	0

Step 7 (Optional) If the Req-For-IP-Filtered column shows the majority of ARP packets, use the **show cable arp-filter ip-requests-filtered** command to display more details about the CPE device that is generating this traffic. Then use the **debug cable mac-address** and **debug cable arp filter** commands to display detailed information about this particular traffic; for example:

Example:

```
Router# show cable arp-filter c5/0/0 ip-requests-filtered 100

Sid  MAC Address      IP Address      Req-Filtered  Req-For-IP-Filtered  Rep-Filtered
1     0007.0e03.1f59  50.3.81.3      0             37282                0
Router# debug cable mac-address 0007.0e03.1f59

Router# debug cable arp filter

Router#
Apr 23 23:03:23.565: ARP for IP Filter=F sid 1 s 0000.0000.0049 d 0005.00e5.3610 sip 50.3.81.13 dip
50.3.82.173 prot 6 len 46 SrcP 445 DstP 445
Apr 23 23:03:23.565: ARP for IP Filter=F sid 1 s 0000.0000.0049 d 0005.00e5.3610 sip 50.3.81.13 dip
50.3.82.174 prot 6 len 46 SrcP 445 DstP 445
Apr 23 23:03:23.565: ARP for IP Filter=F sid 1 s 0000.0000.0049 d 0005.00e5.3610 sip 50.3.81.13 dip
50.3.82.175 prot 6 len 46 SrcP 445 DstP 445
[additional output omitted]...
```

This example shows that the CPE device at IP address 50.3.81.13 is sending packets to TCP port 445 to every IP address on the 50.3.82.0 subnet, in a possible attempt to find a computer that has Microsoft Windows file-sharing enabled.

Step 8 After determining the specific devices that are generating excessive ARP traffic, you can take whatever action is allowed by your company's service level agreements (SLAs) to correct the problem.

Examples

In this example, two cable interfaces, C5/0/0 and C7/0/0, are joined in the same bundle, which means the interfaces share the same broadcast traffic. Separate devices on each interface are generating excessive ARP traffic:

- The device at MAC address 000C.2854.72D7 on interface C7/0/0 is generating or forwarding a large volume of ARP requests. Typically, this device is a cable modem that is forwarding the ARP requests that are being generated by a CPE device behind the modem. The CPE device could be attempting a theft-of-service or denial-of-service attack, or it could be a computer that has been infected by a virus and is trying to locate other computers that can be infected.
- The device at MAC address 000C.53B6.562F on Cable 5/0/0 is responding to a large number of ARP requests, which could indicate that the device is a router that is running faulty software.

The following commands identify the device on the C7/0/0 interface that is generating the excessive ARP requests:

```
Router# show cable arp-filter c7/0/0

ARP Filter statistics for Cable7/0/0:
Replies Rcvd: 3 total. 3 unfiltered, 0 filtered
```

```

Requests Sent For IP: 0 total. 0 unfiltered, 0 filtered
Requests Forwarded: 27906 total. 562 unfiltered, 27344 filtered
Router# show cable arp-filter c7/0/0 requests-filtered 100

```

Sid	MAC Address	IP Address	Req-Filtered	Req-For-IP-Filtered	Rep-Filtered
1	000C.2854.72d7	50.3.81.4	62974	0	0

The following commands identify the device on the C5/0/0 interface that is generating the excessive ARP replies:

```
Router# show cable arp-filter c5/0/0
```

```

ARP Filter statistics for Cable5/0/0:
Replies Rcvd: 2400 total. 456 unfiltered, 1944 filtered
Requests Sent For IP: 0 total. 0 unfiltered, 0 filtered
Requests Forwarded: 26 total. 26 unfiltered, 0 filtered
Router# show cable arp-filter c5/0/0 replies-filtered 100

```

Sid	MAC Address	IP Address	Req-Filtered	Req-For-IP-Filtered	Rep-Filtered
2	000C.53b6.562f	50.3.81.6	0	0	2097

Clearing the Packet Counters

To clear the packet counters on an interface, which includes the ARP packet counters, use the **clear counters cable interface** command. You can also clear the packet counters on all interfaces by using the **clear counters** command without any options. This allows you to use the **show cable arp** commands to display only the CPE devices that are currently generating the most traffic.

The following example shows the ARP packet counters being cleared:

```
Router# show cable arp cable 3/0
```

```

ARP Filter statistics for Cable3/0:
Replies Rcvd: 3278 total. 84 unfiltered, 3194 filtered
Requests Sent For IP: 941 total. 30 unfiltered, 911 filtered
Requests Forwarded: 941 total. 37 unfiltered, 904 filtered

```

```
Router# show cable arp cable 3/0 replies-filtered 1
```

Sid	MAC Address	IP Address	Req-Filtered	Req-For-IP-Filtered	Rep-Filtered
2	0006.2854.71e7	50.3.72.2	1815	0	3194

```
Router# show cable arp cable 3/0 requests-filtered 1
```

Sid	MAC Address	IP Address	Req-Filtered	Req-For-IP-Filtered	Rep-Filtered
2	0006.2854.71e7	50.3.72.2	1815	0	3194

```
Router# clear counter cable 3/0
```

```
Clear "show interface" counters on this interface [confirm] y
```

```
22:38:45.875: %CLEAR-5-COUNTERS: Clear counter on interface Cable3/0 by console
```

```
Router# show cable arp cable 3/0
```

```

ARP Filter statistics for Cable3/0:
Replies Rcvd: 0 total. 0 unfiltered, 0 filtered
Requests Sent For IP: 0 total. 0 unfiltered, 0 filtered
Requests Forwarded: 0 total. 0 unfiltered, 0 filtered

```

```
Router# show cable arp cable 3/0 replies-filtered 1
```

```

Sid  MAC Address      IP Address      Req-Filtered  Req-For-IP-Filtered  Rep-Filtered

Router# show cable arp cable 3/0 requests-filtered 1

Sid  MAC Address      IP Address      Req-Filtered  Req-For-IP-Filtered  Rep-Filtered

```



Note The **clear counters** command clears all of the packet counters on an interface, not just the ARP packet counters.

Identifying ARP Offenders in PXF

When the PXF ARP Filter feature is enabled, use the **sho cable arp-filter interface** command to generate a list of ARP offenders.

The following example shows a list of ARP offenders being generated:

```

Router# show cable arp-filter ?

Bundle  Cable Virtual bundle interface
Cable  CMTS interface
uBR-15#sho cable arp-filter Bundle1 ?
ip-requests-filtered  Show modems with arp request for IP packet filter count
                        at or above x
replies-filtered      Show modems with arp reply filter count at or above x
requests-filtered     Show modems with arp request filter count at or above x
|
Output modifiers
<cr>

```

The following is a sample output from the CLI:

```

Router# show cable arp-filter Bundle1 requests-filtered 40
Interface Cable5/0/0 - none
Interface Cable6/0/2
Sid  MAC Address      IP Address      Req-Filtered  Req-For-IP-Filtered  Rep-Filtered
4    0007.0e03.9cad  50.3.81.15     46            0                    0
Interface Cable7/0/0 - none

```

PRE2 Outputs in PXF

When the PXF ARP Filter feature is enabled, the PRE2 output formatting displays the modem and the CPE addresses on a single line, in addition to the following columns:

- **M/S**—This column shows if packets are being filtered by MAC address or SID. A majority of these columns will show MAC address.
- **Rate**—This column shows the packet rate for PXF-filtered packets in the last 5 minutes monitoring time window. Rate is not calculated for RP-filtered packets.
- **Pro**—This column will identify the processor that performed the filtering with either “RP” or “PXF.” On the PRE2, it is expected that 99.9% of Pro fields will show “PXF.”

The following is a sample output for an ARP request on a PRE2 in PXF:

```
Router# show cable arp-filter Bundle1 requests-filtered 40
Interface Cable5/0/0 - none
Interface Cable6/0/2
Sid  CPE Mac          CPE IP          Modem MAC       Modem IP       M/S Rate Pro REQS
4    00d0.b75a.822a 50.3.81.56      0007.0e03.9cad 50.3.81.15     MAC - RP 46
4    00d0.b75a.822a 50.3.81.56      0007.0e03.9cad 50.3.81.15     MAC 25 PXF 5012
5    00b0.d07c.e51d 50.3.81.57      0007.0e03.1f59 50.3.81.13     MAC - RP 64000
6    -                -                0006.2854.7347 50.3.81.4      MAC 101 PXF 5122
7    -                -                0006.2854.72d7 50.3.81.11     SID - PXF 961205
Interface Cable7/0/0 - none
```

This sample output demonstrates the following:

- SID 4 shows a CPE filtered in PXF. The threshold specified is low enough to show the packets that were filtered on the RP as the offender was being identified. A high enough threshold would not have shown the RP-filtered packets. The ARP packet rate of 25 is shown for PXF-filtered packets.
- SID 5 shows a CPE filtered on the RP. This is extremely unusual and only occurs when the maximum number of PXF-filterable entities has been reached.
- SID 6 shows a modem filtered in PXF (CPE MAC or CPE IP are not shown).
- SID 7 shows ARP packets from an “unknown” source MAC address filtered by SID in PXF.

The counts for requests, replies, and requests for IP will no longer be shown on a single line in order to keep the line concise and less than 90 characters in length.

The “REQs” column is now stated as “REPs” in the case of ARP replies. The column will show “REQ-IP” in cases involving ARP requests for IP.

Requests being sent by the CMTS due to encroaching IP packets, “ip-requests-filtered”, will still be filtered on the RP and not in PXF, with Access Control Lists (ACLs) used to defeat IP-based scanning traffic, and the IP punt rate limiting feature for PRE2 used to decrease the punt rate for such traffic. The ARP Filter can still be used to perform analysis of these IP traffic streams.

PRE1 and Cisco 7246 Outputs in PXF

When the PXF ARP Filter is enabled, the PRE1 and Cisco 7246 output for the show commands is simplified to exclude all columns that do not apply.

The following is a sample output for an ARP request on a PRE1 or 7246 in PXF:

```
Router# show cable arp-filter Bundle1 requests-filtered 40
Interface Cable5/0/0 - none
Interface Cable6/0/2
Sid  CPE Mac          CPE IP          Modem MAC       Modem IP       M/S REQS
4    00d0.b75a.822a 50.3.81.56      0007.0e03.9cad 50.3.81.15     MAC 5058
5    00b0.d07c.e51d 50.3.81.57      0007.0e03.1f59 50.3.81.13     MAC 64000
6    -                -                0006.2854.7347 50.3.81.4      MAC 5122
7    -                -                0006.2854.72d7 50.3.81.11     SID 961205
Interface Cable7/0/0 - none
```

Configuring PXF Divert-Rate-Limit

Use the following procedure to configure Divert-Rate-Limit packet streams to identify potential congestion of the FP-to-RP interface.

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	Do one of the following: <ul style="list-style-type: none"> • service divert-rate-limit divert-code rate limit Example: Router(config)# service divert-rate-limit fib-rp-glean 10 limit 20 Example: Router(config)# service divert-rate-limit fib-rpf-glean 10 limit 20	Configures the Divert-Rate-Limit for the following packets: <ul style="list-style-type: none"> • fwd-glean—Packets that hit a glean adjacency in the FIB. • rpf-glean—Packets that hit a glean adjacency during the RPF check. The rate is the average number of packets-per-second that pass the rate-limiting code. The minimum rate is 1 packet-per-second and the maximum rate is 255 packets-per-second. The default rate is 20 packets-per-second. The minimum limit is 4 packets and the maximum limit is 255 packets. The default limit is 5 packets. Note Using the no form of the service divert-rate-limit command will reset the rate and limit to the default values.
Step 4	end Example: Router(config)# end	Exits interface configuration mode and returns to privileged EXEC mode.

Configuration Examples for Cable ARP Filtering

This section provides the following examples of how to configure the Cable ARP Filtering features:

ARP Filtering Configuration on an Individual Cable Interface: Example

The following example shows a typical configuration of a cable interface that is configured for the Cable ARP Filtering feature:

```

!
interface Cable5/0/0
 ip address 192.168.100.1 255.255.255.0 secondary
 ip address 192.168.110.13 255.255.255.0
 cable downstream annex B
 cable downstream modulation 256qam
 cable downstream interleave-depth 32
 cable downstream channel-id 0
 cable upstream 0 frequency 6000000
 cable upstream 0 power-level 0
 cable upstream 0 channel-width 3200000 200000
 cable upstream 0 minislot-size 16
 cable upstream 0 modulation-profile 6 7
 no cable upstream 0 shutdown
 cable upstream 1 frequency 26000000
 cable upstream 1 power-level 0
 cable upstream 1 channel-width 3200000 200000
 cable upstream 1 minislot-size 4
 cable upstream 1 modulation-profile 6 7
 no cable upstream 1 shutdown
 cable upstream 2 frequency 15008000
 cable upstream 2 power-level 0
 cable upstream 2 channel-width 3200000 200000
 cable upstream 2 minislot-size 4
 cable upstream 2 modulation-profile 6 7
 cable upstream 2 shutdown
 cable upstream 3 spectrum-group 25
 cable upstream 3 channel-width 3200000 200000
 cable upstream 3 minislot-size 4
 cable upstream 3 modulation-profile 1
 cable upstream 3 shutdown
 cable upstream 4 frequency 21008000
 cable upstream 4 power-level 0
 cable upstream 4 channel-width 3200000 200000
 cable upstream 4 minislot-size 16
 cable upstream 4 modulation-profile 1
 no cable upstream 4 shutdown
 cable upstream 5 spectrum-group 25
 cable upstream 5 channel-width 3200000 200000
 cable upstream 5 minislot-size 4
 cable upstream 5 modulation-profile 1
 cable upstream 5 shutdown
 cable arp filter request-send 4 2
 cable arp filter reply-accept 4 2
end

```

ARP Filtering Configuration on Bundled Cable Interfaces: Example

The following example shows a typical configuration of a cable interface bundle that is also using the Cable ARP Filtering feature. Both the master and slave interface are configured separately, allowing you to configure the feature only on the particular interfaces that require it. In addition, you can configure the feature with different threshold values, allowing you to customize the feature for each interface's traffic patterns.

```

!
interface Cable5/0/0
 description Master cable interface
 ip address 10.3.130.1 255.255.255.0 secondary
 ip address 10.3.131.1 255.255.255.0 secondary
 ip address 10.3.132.1 255.255.255.0 secondary

```

```

ip address 10.3.133.1 255.255.255.0 secondary
ip address 10.3.81.1 255.255.255.0
ip helper-address 10.14.0.4
load-interval 30
cable bundle 1 master
cable downstream annex B
cable downstream modulation 64qam
cable downstream interleave-depth 32
cable downstream frequency 441000000
cable downstream channel-id 0
cable upstream 0 frequency 5008000
cable upstream 0 power-level 0
cable upstream 0 channel-width 1600000
cable upstream 0 minislots-size 4
cable upstream 0 modulation-profile 1
no cable upstream 0 shutdown
cable upstream 1 channel-width 1600000
cable upstream 1 minislots-size 4
cable upstream 1 modulation-profile 1
cable upstream 1 shutdown
cable upstream 2 channel-width 1600000
cable upstream 2 minislots-size 4
cable upstream 2 modulation-profile 1
cable upstream 2 shutdown
cable upstream 3 channel-width 1600000
cable upstream 3 minislots-size 4
cable upstream 3 modulation-profile 1
cable upstream 3 shutdown
cable arp filter request-send 4 2
cable arp filter reply-accept 4 2
!
interface Cable7/0/0
description Slave cable interface--Master is C5/0/0
no ip address
cable bundle 1
cable downstream annex B
cable downstream modulation 64qam
cable downstream interleave-depth 32
cable downstream frequency 562000000
cable downstream channel-id 0
no cable downstream rf-shutdown
cable upstream 0 connector 0
cable upstream 0 frequency 5008000
cable upstream 0 power-level 0
cable upstream 0 channel-width 1600000
cable upstream 0 minislots-size 4
cable upstream 0 modulation-profile 21
no cable upstream 0 shutdown
cable upstream 1 connector 1
cable upstream 1 channel-width 1600000
cable upstream 1 minislots-size 4
cable upstream 1 modulation-profile 21
cable upstream 1 shutdown
cable upstream 2 connector 2
cable upstream 2 channel-width 1600000
cable upstream 2 minislots-size 4
cable upstream 2 modulation-profile 21
cable upstream 2 shutdown
cable upstream 3 connector 3
cable upstream 3 channel-width 1600000
cable upstream 3 minislots-size 4
cable upstream 3 modulation-profile 21
cable upstream 3 shutdown
cable arp filter request-send 20 5
cable arp filter reply-accept 20 5
end

```

ARP Filtering in PXF Default Configuration: Example

The following example shows the default configuration of a cable interface for the ARP Filtering in PXF feature.

```
interface Bundle1
 cable arp filter request-send 3 2
 cable arp filter reply-accept 3 2
end
```

Additional References

The following sections provide references related to the Cable ARP Filtering feature.

Related Documents

Related Topic	Document Title
CMTS Commands	<i>Cisco IOS CMTS Cable Command Reference</i> http://www.cisco.com/en/US/docs/ios/cable/command/reference/cbl_book.html

Standards

Standards	Title
SP-RFIV1.1-I09-020830	Data-over-Cable Service Interface Specifications Radio Frequency Interface Specification, version 1.1

MIBs

MIBs	MIBs Link
No new or modified MIBs are supported, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 826	An Ethernet Address Resolution Protocol (ARP)
RFC 2665	DOCSIS Ethernet MIB Objects Support

RFCs	Title
RFC 2669	Cable Device MIB

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 Cable ARP Filtering on the Cisco Cable Modem Termination System

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to <http://tools.cisco.com/ITDIT/CFN/>. An account on <http://www.cisco.com/> is not required.



Note

The below 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.

Table 2: Feature Information for the Cable ARP Filtering Feature

Feature Name	Releases	Feature Information
Cable ARP Filtering	12.2(15)BC2	This feature was introduced for the Cisco uBR7246VXR and Cisco uBR10012 universal broadband routers.

Feature Name	Releases	Feature Information
Cable ARP Filtering	12.2(15)BC2b	The ip-requests-filtered option was added to the service divert-rate-limit command to display the specific Service IDs (SIDs) that are generating or forwarding a minimum number of ARP packets.
Cable ARP Filtering	12.3(9a)BC	Introduced optional syntax for the cable arp filter command, where number and window-size values are optional for reply-accept and request-send settings.
Cable ARP Filtering	12.3(17a)BC	<p>The show cable arp-filter command was introduced for the PXF ARP Filter feature.</p> <p>The service divert-rate-limit command was introduced.</p> <p>Default settings changed for two commands to result as follows:</p> <ul style="list-style-type: none"> • cable arp filter request-send 3 2 • cable arp filter reply-accept 3 2
Cable ARP Filtering	12.2(33)SCA	This feature is integrated into Cisco IOS Release 12.2(33)SCA. Support for the Cisco uBR7225VXR Universal Broadband Router was added.

