

Understanding Traffic Shaping with AIP

Document ID: 10400

Contents

Introduction

Prerequisites

- Requirements

- Components Used

- Conventions

Basic Traffic Shaping

Traffic Shaping with AIP

AIP Features

Burst Size Versus Maximum Burst Size

- Using the Old CLI

- Using the New CLI

AIP Default Behavior

Related Information

Introduction

This document introduces traffic shaping using ATM Interface Processor (AIP) cards and describes the architecture and limitations of these cards.

Note: You do not have to manually assign permanent virtual circuits (PVCs) and switched virtual circuit (SVCs) to rate queues, since more recent versions of the Cisco IOS® Software do this automatically and dynamically. Any references you see to assigning these manually apply only to older versions of the software.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

The information in this document is based on the AIP hardware detailed in the AIP Installation and Configuration Guide. The software version is not relevant except when stated otherwise.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Conventions

For more information on document conventions, refer to the Cisco Technical Tips Conventions.

Basic Traffic Shaping

Non-real time variable bit rate (VBR-nrt) virtual circuits (VCs) are normally configured with a peak rate, average rate and burst size. Each VC specifies a percentage of the peak rate as its average rate. The average rate can either be 100% of the peak rate or a percentage that is less than 50%. The following is an example:

```
atm pvc 6 8 69 aal5snap 512 128 3
```

The example above is a PVC with a peak cell rate of 512 kbps, and a sustainable cell rate of 128 kbps. In this case, the average rate is 25% of the peak rate.

The AIP shapes traffic based on two leaky bucket algorithms. This grants a cell credit to the VC at every service interval corresponding to the average rate.

Note: The total cell credit can not exceed the specified burst size.

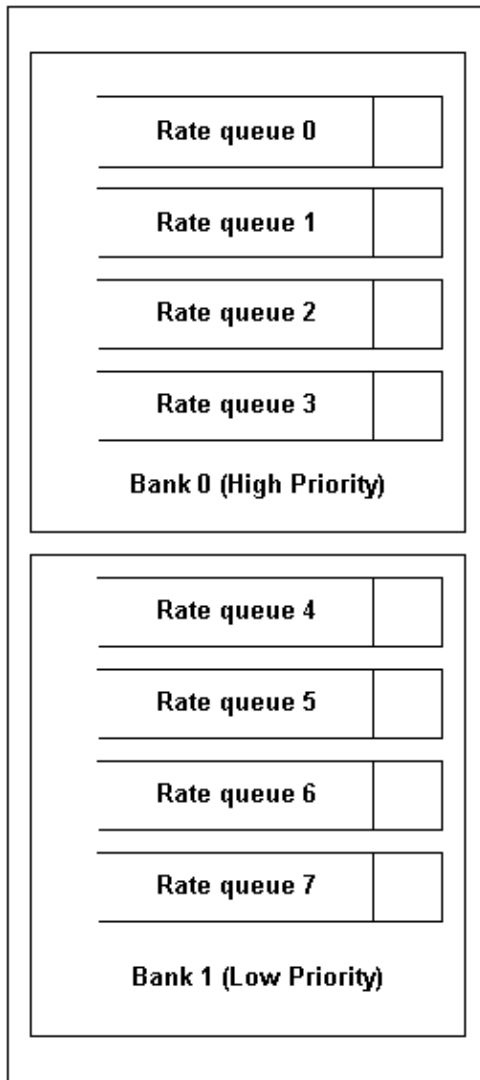
The peak rate of a rate queue determines the service time of that queue. Before transmitting packets, the system software first links them into the corresponding VC structure. It then links this VC structure into the appropriate rate queue. The following section explores this in more detail.

Traffic Shaping with AIP

The ATM segmentation and reassembly (SAR) chip dictates traffic shaping on the AIP. This SAR chip bases its traffic shaping on the notion of rate queues, as described below:

1. Each VC can be allocated a peak rate. This is the maximum rate at which cells can be transmitted on that circuit when there is traffic to send. The system software examines the peak rate of the VC and assigns it to the rate queue that most closely matches the requested rate.
2. Traffic shaping in AIP conforms to ITU-T Traffic Control and Resource Management in B-ISDN. I.371 Recommendation, 1992. I.371 which describes the leaky bucket algorithm. The SAR chip provides eight rate queues for ATM traffic shaping. It groups these eight rate queues into two banks:
 - ◆ Bank zero: rate queues zero through three (0 – 3). This has a higher priority than bank one.
 - ◆ Bank one: rate queues four through seven (4 – 7).
3. The SAR chip maps each VC to a rate queue when it is created. The first VC created uses rate queue zero, the second uses rate queue one, and so on. You can check this using the **show atm interface atm interface number** command. Please refer to the Over-Subscription Problems section later in this document.
4. When you use vbr-nrt, if the peak cell rate (PCR) value is equal to the sustainable cell rate (SCR) value, this is treated as a rate-limited UBR. This feature is documented in Cisco bug ID CSCdm64510 (registered customers only) .

This configuration is not supported in the new Command Line Interface (CLI). For more information on this, click [here](#).



Packets linked to rate queues in the low-priority bank (bank one) cannot transmit while the rate queues in the high-priority bank (bank zero) are not empty.

Although we use priority queuing between the two banks, rate queues within each bank are serviced in a sequential or "round robin" manner. Each VC sends out one cell when the rate queue is served. When a rate queue requests service, the currently-selected VC sends out one cell and the round robin pointer increments to the next VC linked to that rate queue. If two rate queue timers expire at the same time, they are serviced in the round robin fashion, starting from the rate queue with the lower number. As soon as a rate queue has sent out one cell, the service for that queue is complete. There is no traffic policing during reassembly.

Example

If a rate queue is configured as 10 Mbps, when a service opportunity comes, one cell of each VCI in this rate queue is sent out as long as there is a token in its bucket. The service frequency of the rate queue remains constant once configured. As long as the physical layer interface module (PLIM) can handle the speed, every VCI attached to this rate queue is in the peak rate.

This means that if there are only ten virtual channel identifiers (VCIs) on a 10 Mbps rate queue, they can transmit packets at 10 Mbps simultaneously, totaling 100 Mbps.

Over-Subscription Problems

If the system is over-subscribed, this can block the lower priority bank. However, all rate queues in higher priority bank are still serviced.

Over-subscription also has other disadvantages. If we attach 100 VCs to a 5 Mbps queue, this holds the queue for a long time and can, for instance, deprive a 100 Mbps queue that has only one VC. Also, of the 100 VCs attached to this 5 Mbps rate queue, each one may have a different average rate. Therefore, when the 5Mbps rate queue times out and needs to be serviced, not all the VCs have a token in the bucket. This means that fewer than 100 VCIs can be serviced at this time.

Because the request service frequency of the 100 Mbps is much higher than 5 Mbps, the packet can still be sent out. However, this is very slow because the bandwidth is already oversubscribed . In the worst case scenario, the other queue can be totally deprived.

AIP Features

There are three parameters used to manage the AIP traffic flow:

- Peak rate
- Average rate
- Burst

The PCR determines which rate-queue the VCD will be attached to and determines the service time of that rate-queue. PCR will be maintained as long as the VC's SCR bucket has credits. The average rate determines the time period for one token to put into the bucket. The average rate determines the SCR. Credits accumulate at a rate equal to SCR.

The AIP Sat chip set requires SCR & PCR to be linked by the following formula:

$$SCR = 1/n * PCR \quad (n=1\&.64)$$

The burst size determines the maximum number of token to be put into the bucket. The total credit cannot exceed the specified burst size. The burst size ranges from 0 –63. The rate queue is serviced at the rate equal to PCR . Therefore if a VC has constant data to send it will send only at the rate equal to SCR and will not burst. If the amount of data falls below the SCR then credits will start accumulating up to the burst size. If the amount of data to send out VC increased, a burst equal to the burst size can be sent by VC. After the burst the data can again be sent at the SCR rate.

Here are the key features of the AIP:

- Peak Rate Range: 155 Mbps down to 130 Kbps.
- Sustained Rate : $SCR = 1/n * PCR$ (where n is an integer and $n=1\&.64$)

Note: You can also set the SCR to be the same as the PCR.

- With the old CLI, you cannot set the burst size to zero, since it is a multiple of 32 cells.

For example, `atm pvc 6 8 69 aa15snap 256 128 3` means you are using 3 x 32 cells as the burst size (96 cells).

- The VCI range can be set from zero to 65535.

Burst Size Versus Maximum Burst Size

Depending on the way we have configured the PVC with VBR-nrt, the parameter used to configure the amount of cells sent at PCR changes.

Using the Old CLI

If you use the old CLI, the parameter configured is not the Maximum Burst Size (MBS) but the burst size. This burst size is a multiple of 32 cells.

```
router(config-subif)#atm pvc 6 8 69 aal5snap 256 128 ?
<1-63>  Burst size in number of 32 cell bursts
inarp   Inverse ARP enable
oam     OAM loopback enable
<cr>
```

For example, the command shown here (**atm pvc 6 8 69 aal5snap 256 128 3**) means you are using 3 x 32 cells as the burst size (96 cells). This burst size is the parameter the AIP uses in its shaping algorithm. It does not represent the amount of cells that are really sent at PCR.

Let's look at the relationship between the configured burst size and the MBS found in VBR-nrt. These two parameters are linked by the following formula:

$$\text{MBS} = \text{number of cells at PCR} = [(\text{BURST SIZE} \times 32 \times 424) / (\text{PCR} - \text{SCR})] * [\text{PCR} / 424]$$

The PCR and the SCR we are using in the formula above are not the configured values, but the values that the AIP uses to do the traffic shaping. This problem is due to the AIP shaper granularity. Let's look at an example to illustrate this:

```
interface ATM1/0.5 point-to-point
 atm pvc 7 10 500 aal5snap 5000 2500 52

router#show atm vc
```

Interface	Name	VPI	VCI	Type	Encaps	SC	Peak Kbps	Avg/Min Kbps	Burst Cells	Sts
1/0.5	7	10	500	PVC	SNAP	VBR	5000	2500	3264	UP

As we can see here, the configured burst size is equal to 1664 cells (52 x 32) but the actual MBS is equal to 3264 cells.

Using the New CLI

When using the new CLI (in Cisco IOS Software Releases 12.0 and above), the parameter configured is the MBS and not the burst size as we saw in the previous section. The router still internally converts the configured MBS into a burst size used in its shaping algorithm. Since the MBS is still linked to the burst size through the formula shown in the previous section, the MBS which could be measured on the outgoing traffic might still differ slightly from the configured value.

The difference is that this operation is now transparent to the user who configures what he needs (in other words, the MBS).

Here is an example illustrating this behavior with the new CLI:

```
router(config)#interface ATM1/0.3 point-to-point
router(config-subif)#pvc 10/300
router(config-if-atm-vc)#vbr-nrt 5000 2500 ?
```

```

<64-4032> Maximum Burst Size(MBS) in Cells
<cr>

router(config-if-atm-vc)#vbr-nrt 5000 2500 1000
router(config-if-atm-vc)#^Z
router#sh atm vc
          VCD /
Interface  Name          VPI   VCI  Type  Encaps  SC   Peak  Avg/Min  Burst  Sts
1/0.3     5                10   300  PVC   SNAP    VBR   5000  2500    960   UP

```

As you can see in the output above, the user can now directly configure the desired MBS but due to the AIP granularity, the real MBS might be slightly different from the configured MBS.

AIP Default Behavior

If you leave the burst size undefined, the AIP takes three as the default value. For example:

```
atm pvc 6 8 69 aal5snap 256 128
```

is equivalent to:

```
atm pvc 6 8 69 aal5snap 256 128 3
```

You can set the SCR to be the PCR value divided by n ($SCR = 1/n * PCR$ (where n is an integer and $n=1\&.64$).

If you set $SCR=PCR/n$ where n is not an integer, the AIP rounds up the value without displaying an error. The AIP also lets you specify values under $PCR/2$, then rounds them up without notifying you. For example, if you type:

```
atm pvc 6 8 69 aal5snap 512 200 1 (where the SCR is equal to PCR divided by 2.56)
```

the AIP interprets this as:

```
atm pvc 6 8 69 aal5snap 512 256 1 (where the SCR is rounded up to PCR divided by 2)
```

The AIP rounds this figure up to a higher value. In all cases, you are recommended to use an integer for n .

Related Information

- [ATM Traffic Management Technology Support](#)
- [ATM Technology Support](#)
- [Broadband Forum](#) 
- [Technical Support & Documentation – Cisco Systems](#)

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2013 – 2014 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

Updated: Nov 15, 2007

Document ID: 10400
