

# Understanding PVC Bundle Bumping and Protection

Document ID: 10491

---

## Introduction

### Prerequisites

- Requirements
- Components Used
- Conventions

### Circuit Bumping

- Implicit Bumping
- Explicit Bumping
- Additional Bumping Information
- Bring the PVC Down
- Bundle's Protected Group
- Individually-protected VC Bundle Member

### debug and show Commands

### Related Information

---

## Introduction

ATM Permanent Virtual Circuit (PVC) bundles allow you to configure multiple PVCs that have different Quality of Service (QoS) characteristics between two devices. The purpose is to bind a PVC from the bundle to one or more precedence values. To determine which VC in the bundle is used to forward specific traffic, the ATM VC bundle management software matches precedence levels between packets and VCs.

The ATM virtual bundle acts as a single routing link to the destination router. The Operation and Maintenance (OAM) polling mechanism individually monitors the integrity of each circuit individually. If a VC becomes unavailable, the router can avoid losing data in two ways:

- Circuit Bumping
- Bring the whole bundle down

The purpose of this document is to explain in detail the concepts behind PVC bundle protection. Refer to PVC Bundle Protection for more information.

## Prerequisites

## Requirements

There are no specific requirements for this document.

## Components Used

The information in this document is based on these software and hardware versions:

- 7500:
  - ◆ All PA-A3 versions. Requires VIP2-50, VIP4-50, VIP4-80 or later. Only one PA-A3 per

- ◆ VIP2–50. Requires Cisco IOS® Software Release 12.0(3)T and later.
  - ◆ All PA–A6 versions. Requires VIP4, RSP8 and later. Requires Cisco IOS Software Release 12.2(15)T, 12.3(1) and later.
- 7200/7200VXR:
  - ◆ All PA–A3 versions. NPE200 or higher, Requires Cisco IOS Software Release 12.0(3)T and later.
  - ◆ All PA–A6 versions. 7200VXR: NPE400, NSE–1, NPE–G1. Requires Cisco IOS Software Release 12.2(15)T, 12.3(1) and later.
- Catalyst 6500/7600:
  - ◆ FlexWan and Enhanced FlexWan with PA–A3. Supported with MSFC2 in Cisco IOS Software Release 12.1(13)E13 and later.
  - ◆ FlexWan and Enhanced FlexWan with PA–A6. Supported on SUP2 and SUP720 in Cisco IOS Software Release 12.2(14)SX and later.
- 7401ASR:
  - ◆ All PA–A6 versions. Requires Cisco IOS Software Release 12.3(1) and later.
  - ◆ All PA–A3 versions. Supported in all Cisco IOS Software releases.
- Cisco 2600 and 3600:
  - ◆ NM–1A–OC3, NM–4E1–IMA, NM–4T1–IMA, NM–8E1–IMA, NM–8T1–IMA network modules. Requires Cisco IOS Software Release 12.0(7)T or later.
  - ◆ NM–1A–T3 and NM–1A–E3 network modules. Requires Cisco IOS Software Release 12.1(2)T or later.
- Cisco 12000:
  - ◆ Cisco 12000 8–port OC–3c/STM1 ATM Line Card. Cisco IOS Software Release 12.0(23)S or later.
  - ◆ Cisco 12000 4 –port OC–3c/STM–1c ATM ISE Line Card. Cisco IOS Software Release 12.0(27)S or later.

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

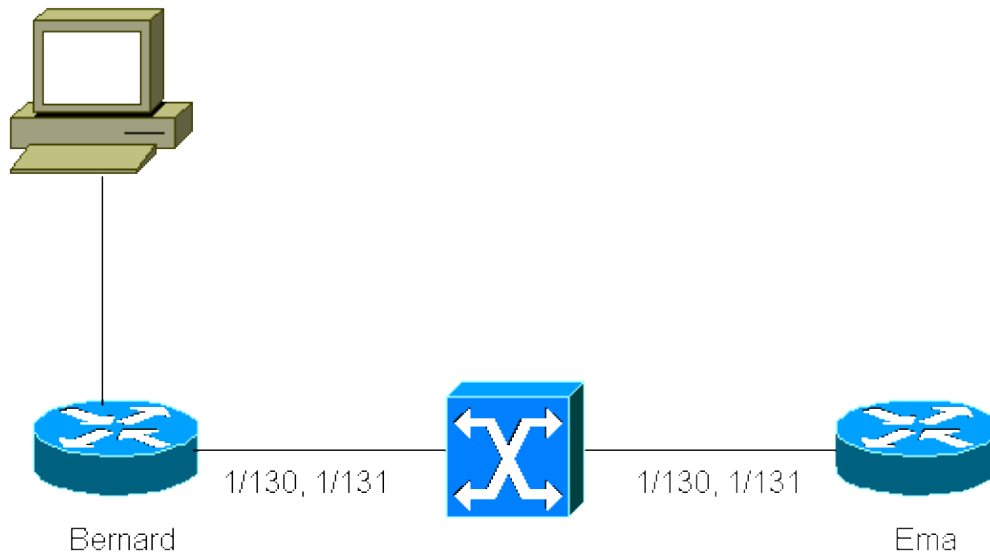
For more information on document conventions, refer to the Cisco Technical Tips Conventions.

## Circuit Bumping

Circuit bumping assigns the traffic bound to a failed VC to another VC. The two bumping methods are explicit bumping and implicit bumping.

In order to illustrate bumping, this setup is used:

Traffic Generator



Two PVCs are created between Ema and Bernard (two 7507s that use a PA-A3 in a VIP2-50 and run Cisco IOS Software Release 12.0(7)T). These two PVCs are assigned the values 1/130 and 1/131 on both ends of the connection. To simplify, 1/130 is switched to 1/130 by the ATM switch and 1/131 is switched to 1/131.

The PVC 1/130 is configured as a VBR-nrt VC and 1/131 as an ABR VC. The precedence values 0 to 4 are bound to PVC 1/130 and precedence values 5 to 7 to the PVC 1/131. The Per-VC DWRED is used to discard packets.

## Implicit Bumping

In case there is a VC failure, implicit bumping (which is the default) diverts the traffic to a VC in charge of a lower precedence value. Here are the configuration lines necessary to enable this protection method (configuration taken from Bernard, similar to the configuration that exists on Ema) if the traffic generator sends two streams of data, one with precedence 5 and one with precedence 3:

```
interface ATM2/0/0.6 point-to-point
 bundle bernard
  oam-bundle manage
  pvc-bundle 1/131
    random-detect attach testWRED
    precedence 5-7
    bump implicit
    bump traffic
  pvc-bundle 1/130
    random-detect attach testWRED
    precedence 0-4
    bump implicit
    bump traffic
```

The PVC 1/131 is brought down and this should cause the precedence 5 traffic to go over 1/130.

## Normal Situation

```
bernard#show atm bundle
```

```
bernard on ATM2/0/0.6: UP
```

VC Name	VPI/ VCI	Config Preced.	Current Preced.	Bumping Preced./	PG/ PV	Peak Kbps	Avg/Min kbps	Burst Cells	Sts
---------	----------	----------------	-----------------	------------------	--------	-----------	--------------	-------------	-----

Accept

3	1/131	7-5	7-5	4 / Yes	-	1000	100		UP
6	1/130	4-0	4-0	- / Yes	-	100000	100000	94	UP

**Note:** Go to the Additional Bumping section for an explanation of the output above.

```
bernard#show queuing interface atm 2/0/0.6
```

```
Interface ATM2/0/0.6 VC 1/131
```

```
Exp-weight-constant: 2 (1/4)
```

```
Mean queue depth: 781
```

```
Queue size: 776 Maximum available buffers: 2628
```

```
Output packets: 1309 WRED drops: 217 No buffer: 73982
```

Class	Random drop	Tail drop	Minimum threshold	Maximum threshold	Mark probability	Output Packets
0	0	0	20	40	1/10	0
1	0	0	22	40	1/10	0
2	0	0	24	40	1/10	0
3	0	0	100	1000	1/3	0
4	0	0	28	40	1/10	0
<b>5</b>	<b>217</b>	<b>0</b>	<b>200</b>	<b>1000</b>	<b>1/5</b>	<b>2616</b>
6	0	0	32	40	1/10	0
7	0	0	34	40	1/10	0

```
Interface ATM2/0/0.6 VC 1/130
```

```
Exp-weight-constant: 2 (1/4)
```

```
Mean queue depth: 0
```

```
Queue size: 0 Maximum available buffers: 2628
```

```
Output packets: 3037 WRED drops: 0 No buffer: 72478
```

Class	Random drop	Tail drop	Minimum threshold	Maximum threshold	Mark probability	Output Packets
0	0	0	20	40	1/10	0
1	0	0	22	40	1/10	0
2	0	0	24	40	1/10	0
<b>3</b>	<b>0</b>	<b>0</b>	<b>100</b>	<b>1000</b>	<b>1/3</b>	<b>3037</b>
4	0	0	28	40	1/10	0
5	0	0	200	1000	1/5	0
6	0	0	32	40	1/10	0
7	0	0	34	40	1/10	0

Traffic flows on the appropriate VC. In other words, IP precedence 3 traffic flows across PVC 1/130 and IP precedence 5 traffic flows through PVC 1/131 as configured.

**At this point, PVC 1/131 is brought down.**

As you can see, there are no more packets on 1/131.

```
bernard#show queuing interface atm 2/0/0.6
```

```
Interface ATM2/0/0.6 VC 1/131
```

```
Exp-weight-constant: 2 (1/4)
```

```
Mean queue depth: 779
```

```
Queue size: 778 Maximum available buffers: 2628
```

```
Output packets: 2618 WRED drops: 436 No buffer: 147848
```

Class	Random drop	Tail drop	Minimum threshold	Maximum threshold	Mark probability	Output Packets
0	0	0	20	40	1/10	0
1	0	0	22	40	1/10	0
2	0	0	24	40	1/10	0
3	0	0	100	1000	1/3	0
4	0	0	28	40	1/10	0
<b>5</b>	<b>436</b>	<b>0</b>	<b>200</b>	<b>1000</b>	<b>1/5</b>	<b>2616</b>

6	0	0	32	40	1/10	0
7	0	0	34	40	1/10	0

Interface ATM2/0/0.6 VC 1/130

Exp-weight-constant: 2 (1/4)

Mean queue depth: 264

Queue size: 265 Maximum available buffers: 2628

Output packets: 36666 WRED drops: 657 No buffer: 164145

Class	Random drop	Tail drop	Minimum threshold	Maximum threshold	Mark probability	Output Packets
0	0	0	20	40	1/10	0
1	0	0	22	40	1/10	0
2	0	0	24	40	1/10	0
<b>3</b>	<b>498</b>	<b>0</b>	<b>100</b>	<b>1000</b>	<b>1/3</b>	<b>23252</b>
4	0	0	28	40	1/10	0
<b>5</b>	<b>159</b>	<b>0</b>	<b>200</b>	<b>1000</b>	<b>1/5</b>	<b>13679</b>

*!--- IP precedence 5 traffic is redirected.*

6	0	0	32	40	1/10	0
7	0	0	34	40	1/10	0

In this case, as PVC 1/131 has been brought down, the stream with IP precedence = 5 has been re-directed to PVC 1/130. In other words, IP precedence = 5 traffic is sent across a PVC which is responsible for a lower precedence value. The same happens to traffic with an IP precedence that ranges from 5 to 7.

**Important!:** With such a method, the traffic that corresponds to IP precedence = 0 is not diverted to another VC and is therefore lost. To protect such traffic, use the explicit bumping method.

## Explicit Bumping

In case of a VC failure, you can force the traffic of a failed VC to a specific one rather than let a stream that corresponds to an IP precedence automatically go to a VC that carries a lower IP precedence value. In other words, you manually specify the precedence level to which traffic on a VC (bundle-vc mode) is bumped when the VC goes down. This is called explicit bumping.

The command to enable explicit bumping is **bump explicit [precedence value]**. Here are the configuration lines necessary to enable this protection method (configuration taken from Bernard, similar to the Ema configuration). In this case, PVC 1/130 is configured for explicit bumping while PVC 1/131 keeps the implicit bumping method:

```
interface ATM2/0/0.6 point-to-point
bundle bernard
  oam-bundle manage
  pvc-bundle 1/131
    random-detect attach testWRED
    precedence 5-7
    bump implicit
    bump traffic
  pvc-bundle 1/130
    random-detect attach testWRED
    precedence 0-4
    bump explicit 6
    bump traffic
```

This configuration means that if PVC 1/130 fails, the traffic of this VC is carried by the VC in charge of carrying IP precedence 6 traffic.

To show the results, the traffic generator sends two streams of data, one with precedence 5 and one with precedence 3. The next section discusses what happens on Bernard if PVC 1/130 fails.

**Normal situation, PVC 1/130 is up.**

```
bernard#show atm bundle
```

```
bernard on ATM2/0/0.6: UP
```

VC Name	VPI/ VCI	Config Preced.	Current Preced.	Bumping Preced./ Accept	PG/ PV	Peak Kbps	Avg/Min kbps	Burst Cells	Sts
3	1/131	7-5	7-0	- / Yes	-	1000	100		UP
6	1/130	4-0		6 / Yes	-	100000	100000	94	UP

The **show atm bundle** command allows you to see the bumping method that you configured. Here you can see that PVC 1/130 has been configured for explicit bumping to the PVC in charge of IP precedence **6**.

```
bernard#show queuing interface atm 2/0/0.6
```

```
Interface ATM2/0/0.6 VC 1/131
```

```
Exp-weight-constant: 2 (1/4)
```

```
Mean queue depth: 780
```

```
Queue size: 493 Maximum available buffers: 2628
```

```
Output packets: 10159 WRED drops: 274 No buffer: 43004
```

Class	Random drop	Tail drop	Minimum threshold	Maximum threshold	Mark probability	Output Packets
0	0	0	20	40	1/10	0
1	0	0	22	40	1/10	0
2	0	0	24	40	1/10	0
3	0	0	100	1000	1/3	0
4	0	0	28	40	1/10	0
<b>5</b>	<b>273</b>	<b>0</b>	<b>200</b>	<b>1000</b>	<b>1/5</b>	<b>2123</b>
6	0	0	32	40	1/10	0
7	0	0	34	40	1/10	0

```
Interface ATM2/0/0.6 VC 1/130
```

```
Exp-weight-constant: 2 (1/4)
```

```
Mean queue depth: 0
```

```
Queue size: 0 Maximum available buffers: 2628
```

```
Output packets: 5090 WRED drops: 0 No buffer: 40314
```

Class	Random drop	Tail drop	Minimum threshold	Maximum threshold	Mark probability	Output Packets
0	0	0	20	40	1/10	0
1	0	0	22	40	1/10	0
2	0	0	24	40	1/10	0
<b>3</b>	<b>0</b>	<b>0</b>	<b>100</b>	<b>1000</b>	<b>1/3</b>	<b>5090</b>
4	0	0	28	40	1/10	0
5	0	0	200	1000	1/5	0
6	0	0	32	40	1/10	0
7	0	0	34	40	1/10	0

Each PVC carries its own traffic.

**This output brings down PVC 1/130.**

```
bernard#show atm bundle
```

```
bernard on ATM2/0/0.6: UP
```

```
Config Current Bumping PG/ Peak Avg/Min Burst
```

VC Name	VPI/ VCI	Preced.	Preced.	Preced./ PV Accept	Kbps	kbps	Cells	Sts
3	1/131	7-5	7-0	- / Yes -	1000	100		UP
6	1/130	4-0		6 / Yes -	100000	100000	94	DOWN

bernard#show queuing interface atm 2/0/0.6

Interface ATM2/0/0.6 VC 1/131

Exp-weight-constant: 2 (1/4)

Mean queue depth: 780

Queue size: 783 Maximum available buffers: 2628

Output packets: 12332 WRED drops: 733 No buffer: 152725

Class	Random drop	Tail drop	Minimum threshold	Maximum threshold	Mark probability	Output Packets
0	0	0	20	40	1/10	0
1	0	0	22	40	1/10	0
2	0	0	24	40	1/10	0
<b>3</b>	<b>289</b>	<b>0</b>	<b>100</b>	<b>1000</b>	<b>1/3</b>	<b>1189</b>

*!--- IP precedence 3 traffic has been redirected.*

4	0	0	28	40	1/10	0
<b>5</b>	<b>443</b>	<b>0</b>	<b>200</b>	<b>1000</b>	<b>1/5</b>	<b>3381</b>
6	0	0	32	40	1/10	0
7	0	0	34	40	1/10	0

Interface ATM2/0/0.6 VC 1/130

Exp-weight-constant: 2 (1/4)

Mean queue depth: 0

Queue size: 0 Maximum available buffers: 2628

Output packets: 5090 WRED drops: 0 No buffer: 40314

Class	Random drop	Tail drop	Minimum threshold	Maximum threshold	Mark probability	Output Packets
0	0	0	20	40	1/10	0
1	0	0	22	40	1/10	0
2	0	0	24	40	1/10	0
<b>3</b>	<b>0</b>	<b>0</b>	<b>100</b>	<b>1000</b>	<b>1/3</b>	<b>5090</b>
4	0	0	28	40	1/10	0
5	0	0	200	1000	1/5	0
6	0	0	32	40	1/10	0
7	0	0	34	40	1/10	0

IP precedence 3 traffic now flows through PVC 1/131. This traffic has been explicitly bumped to the PVC in charge of carrying IP precedence = 6 traffic. The advantage of this method is that you can choose which VC carries the traffic of a failed VC. In other words, this allows you to choose a VC which provides a QoS similar to the one that the failed VC provided.

## Additional Bumping Information

When you configure bumping, you can also make sure that bumped traffic does not have an impact on sensitive traffic on another PVC. It is possible for you to make sure that certain VCs never carry bumped traffic. This can be achieved by the use of the **no bump traffic** command.

```
interface ATM2/0/0.6 point-to-point
 bundle bernard
  pvc-bundle 1/130
  bump explicit 6
  no bump traffic
```

The configuration example above means that PVC 1/130 bumps its traffic to the PVC in charge of IP precedence 6. Furthermore, this PVC does not accept bump traffic that comes from another PVC. You can verify the bumping configuration using the **show atm bundle** command:

**Note:** This output is not related to the setup used for previous examples.

```
bernard#show atm bundle

bernard on ATM2/0/0.6: UP

VC Name          VPI/ VCI      Config      Current      Bumping      PG/ Peak  Avg/Min  Burst
                  VCI          Preced.     Preced.     Preced./    PV  Kbps   kbps    Cells  Sts
                  VCI          Preced.     Preced.     Accept
                  VCI          Preced.     Preced.     Accept
                  VCI          Preced.     Preced.     Accept

8                1/132
7-5
4                4 / No      - 149760      DOWN
9                1/131
4-2
7-2              1 / Yes     - 1000        100         UP
10               1/130
1-0
1-0              2 / Yes     - 100000     100000     94         UP
```

These conclusions are made from the **show atm bundle** output:

- You can see the *precedence value(s)* assigned to each PVC.
- You can see to which **precedence** (in the blue font) the traffic of a VC is bumped.
- You can also see whether a PVC **does or does not accept** bumped traffic.

From this output, you can observe that a failure occurred (**PVC 1/132 DOWN**). Due to this failure, PVC 1/131 *now transports traffic* which was previously assigned to PVC 1/132. In other words, PVC 1/131 now transports traffic *with IP precedence between 2 and 7*.

## Bring the PVC Down

When a failure occurs on a VC, there is a second method of protection. This time, instead of redirecting the affected traffic onto another VC, the bundle is brought down. By doing this, you rely on the routing protocol to re-direct the affected traffic onto another connection.

The two methods are bundle's protected group or individually protected VC bundle member. In order to illustrate this, this setup is used:

Traffic Generator



Three PVCs are created between Ema and Bernard (two 7507s that use a PA-A3 in a VIP2-50 and run Cisco IOS Software Release 12.0(7)T). These three PVCs have been assigned the values 1/129, 1/130 and 1/131 on both ends of the connection. For the sake of clarity, 1/129 is switched to 1/129 by the ATM switch, 1/130 is switched to 1/130, and 1/131 is switched to 1/131.

The PVC 1/129 is a simple UBR PVC, 1/130 is configured as a VBR-nrt VC, and 1/131 as an ABR VC. The precedence values 1 to 4 are bound to PVC 1/130, precedence values 5 to 7 to the PVC 1/131, and precedence 0 to 1/129. Per-VC DWRED is used as a packet discard mechanism.

## Bundle's Protected Group

In this method, you assign a VC into a protected group. If all the VCs which are part of a protected group go down, the entire bundle goes down. Here are the configuration lines necessary to enable this protection method (configuration taken from Bernard, a similar configuration exists on Ema). In this case, PVC 1/130 and PVC 1/131 are members of one protected group:

```
interface ATM2/0/0.6 point-to-point
 bundle bernard
  oam-bundle manage
  pvc-bundle 1/131
   protect group
  pvc-bundle 1/130
   protect group
```

This is what you observe when a failure occurs on a VC that is part of such a protected group:

### Normal Situation

```
bernard#show atm bundle
```

```
bernard on ATM2/0/0.6: UP
```

VC Name	VPI/ VCI	Config Preced.	Current Preced.	Bumping Preced./ Accept	PG/ PV	Peak Kbps	Avg/Min kbps	Burst Cells	Sts
8	1/131	7-5	7-5	0 / No	<b>PG</b>	1000	100		UP
9	1/130	4-1	4-1	0 / No	<b>PG</b>	100000	100000	94	UP

```
10*          1/129  0          0          - / No  - 149760          UP
```

You can verify here that PVC 1/130 and 1/131 are part of a **protected group**.

### PVC 1/131 is brought down.

```
bernard on ATM2/0/0.6: UP
```

VC Name	VPI/ VCI	Config Preced.	Current Preced.	Bumping Preced./ Accept	PG/ PV	Peak Kbps	Avg/Min kbps	Burst Cells	Sts
8	1/131	7-5		0 / Yes	PG	1000	100		<b>DOWN</b>
9	1/130	4-1	4-1	0 / Yes	PG	100000	100000	94	UP
10*	1/129	0	7-5, 0	- / Yes	-	149760			UP

PVC 1/131 is down, but PVC 1/130 is up. Therefore, the bundle stays up and traffic from PVC 1/131 is normally bumped to 1/129. The bumping method is the explicit bumping to the *PVC in charge of IP precedence = 0*.

### PVC 1/131 and 1/130 are brought down.

```
bernard#show atm bundle
```

```
bernard on ATM2/0/0.6: DOWN, PG down
```

VC Name	VPI/ VCI	Config Preced.	Current Preced.	Bumping Preced./ Accept	PG/ PV	Peak Kbps	Avg/Min kbps	Burst Cells	Sts
8	1/131	7-5		0 / Yes	PG	1000	100		DOWN
9	1/130	4-1		0 / Yes	PG	100000	100000	94	DOWN
10*	1/129	0		- / Yes	-	149760			UP

```
bernard#show interface atm 2/0/0.6
```

```
ATM2/0/0.6 is down, line protocol is down
```

```
Hardware is cyBus ENHANCED ATM PA
Internet address is 14.0.0.1/8
MTU 4470 bytes, BW 149760 Kbit, DLY 80 usec,
  reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ATM
72 packets input, 6640 bytes
70 packets output, 6560 bytes
1137 OAM cells input, 1296 OAM cells output
```

In this case, both PVCs that belong to the protection group are down. This caused the bundle to go down (**DOWN, PG down**) along with the sub-interface.

**Note:** When you configure bundle protection, you have to remember to also configure bumping on the PVC's part of the protection group. Also, if one of these PVCs goes down but the bundle stays up, the traffic of the failed PVC has to be re-directed to another PVC to keep the integrity of the affected streams. If bumping is not configured, the traffic on the failed PVC cannot be forwarded. This causes the bundle to go down.

This output shows that PVC 1/131 is brought down, PVC 1/130 is up (so the bundle should remain up), but bumping is not configured:

```
bernard#show atm bundle
```

```
bernard on ATM2/0/0.6: DOWN, Bumping exhausted
```

Config	Current	Bumping	PG/	Peak	Avg/Min	Burst
--------	---------	---------	-----	------	---------	-------

VC Name	VPI/ VCI	Preced.	Preced.	Preced./ PV Accept	Kbps	kbps	Cells	Sts
8	1/131	7-5		- / No	PG 1000	100		<b>DOWN</b>
9	1/130	4-1		- / No	PG 100000	100000	94	UP
10*	1/129	0		- / No	- 149760			UP

As you can see, the whole bundle has been brought down.

## Individually-protected VC Bundle Member

In this case, if an individually-protected VC goes down, the whole bundle is brought down. Here are the configuration lines necessary to enable this protection method (configuration taken from Bernard, a similar configuration exists on Ema). In this case, PVC 1/130 is the individually-protected VC:

```
interface ATM2/0/0.6 point-to-point
bundle bernard
  oam-bundle manage
  pvc-bundle 1/130
  protect vc
```

This is what you can observe when a failure occurs on PVC 1/130:

### Normal situation:

```
bernard#show atm bundle
```

```
bernard on ATM2/0/0.6: UP
```

VC Name	VPI/ VCI	Config Preced.	Current Preced.	Bumping Preced./ Accept	PG/ PV	Peak Kbps	Avg/Min kbps	Burst Cells	Sts
8	1/131	7-5	7-5	- / No	-	1000	100		UP
9	1/130	4-1	4-1	- / No	<b>PV</b>	100000	100000	94	UP
10*	1/129	0	0	- / No	-	149760			UP

You can see that PVC 1/130 is configured as a **protected VC**.

### PVC 1/130 is brought down.

```
bernard#show atm bundle
```

```
bernard on ATM2/0/0.6: DOWN, PV down
```

VC Name	VPI/ VCI	Config Preced.	Current Preced.	Bumping Preced./ Accept	PG/ PV	Peak Kbps	Avg/Min kbps	Burst Cells	Sts
8	1/131	7-5		- / No	-	1000	100		UP
9	1/130	4-1		- / No	<b>PV</b>	100000	100000	94	<b>DOWN</b>
10*	1/129	0		- / No	-	149760			UP

```
bernard#show interface atm 2/0/0.6
```

```
ATM2/0/0.6 is down, line protocol is down
```

```
Hardware is cyBus ENHANCED ATM PA
Internet address is 14.0.0.1/8
MTU 4470 bytes, BW 149760 Kbit, DLY 80 usec,
  reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ATM
86 packets input, 7880 bytes
82 packets output, 7700 bytes
```

1465 OAM cells input, 1666 OAM cells output

As you can see, the whole bundle and the corresponding sub-interface has been brought down.

## debug and show Commands

Certain **show** commands are supported by the Output Interpreter Tool ( registered customers only) , which allows you to view an analysis of **show** command output.

**Note:** Before issuing **debug** commands, refer to Important Information on Debug Commands.

- **show atm bundle** *[bundle name]* Displays the bundle attributes assigned to each bundle virtual circuit (VC) member and the current working status of the VC members.
- **show atm pvc** *vpi/vci* Displays all ATM permanent virtual connections (PVCs) and traffic information as well as ATM virtual path identifier (VPI) and virtual channel identifier (VCI) numbers.
- **debug atm bundle events** Displays switched virtual circuit (SVC) bundle events.
- **debug atm bundle errors** Displays debug messages for SVC bundle errors.

---

## Related Information

- [PVC Bundle Protection](#)
- [ATM Acronyms](#)
- [ATM Technology Support Pages](#)
- [Technical Support – Cisco Systems](#)

---

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

© 2008 – 2009 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

---

Updated: Nov 15, 2007

Document ID: 10491

---