Technical Guide

Priority-based Flow Control (PFC)

Feature Overview and Configuration Guide

Introduction

This guide describes Priority-based Flow Control (PFC) and how to configure it.

Priority-based Flow Control (PFC) is an enhancement to traditional Ethernet flow control. It allows per-priority pause of traffic rather than pausing all traffic on a link. This lets you provide lossless Ethernet for specific traffic classes (e.g., storage or real-time traffic) by enabling flow control on a per-priority basis.

Contents

Products and software version that apply to this guide

This guide applies to AlliedWare Plus™ products that support Priority-based Flow Control (PFC), running version **5.5.5-2** or later.

The lossless egress pool and Data Center Bridging eXchange protocol (DCBX) features are available from version **5.5.6-0** or later.

To see whether your product supports Priority-based Flow Control (PFC), see the following documents:

- The product's Datasheet
- The product's Command Reference

Related documents

The following documents give more information about the Priority-based Flow Control (PFC) features on AlliedWare Plus products:

- The product's Command Reference

These documents are available from the links above or on our website at alliedtelesis.com

Pause frames and priority-based flow control (PFC)

Modern data centers and enterprise networks rely heavily on Ethernet for high-speed communication. However, traditional Ethernet does not guarantee lossless transmission. This can be a problem for applications that are sensitive to packet loss, such as storage traffic (e.g., iSCSI, RoCE) or real-time data streams.

Priority-based Flow Control (PFC) addresses this issue by enabling selective pausing of traffic based on priority. Unlike traditional flow control mechanisms that pause all traffic on a link, PFC allows network devices to pause specific classes of traffic while allowing others to continue. This fine-grained control helps prevent congestion-related packet loss without compromising overall throughput.

If your network carries mixed traffic types, such as storage, voice, and general data, PFC can help ensure that critical traffic flows are protected from congestion, improving performance and reliability.

Egress queues and class of service (CoS)

Each interface has a number of egress queues, numbered 0-7. These allow you to prioritize outgoing traffic. By default, these queues are mapped to Class of Service (CoS). CoS is an Ethernet frame header field that specifies a value for the frame, also between 0-7.

By default, the egress queues are configured as strict priority queues. This means they will prioritize frames that match the mapped CoS value, and handle them from highest to lowest. This means all

traffic must have egressed out of a higher priority queue before traffic in a lower priority queue is forwarded. For example, if we configure queues 7 and 6 as strict priority queues, queue 7 must be empty before any traffic in queue 6 is forwarded.

Queues can also be configured as part of a weighted round-robin (WRR) group. With WRR queues, traffic egresses the queue according to a set weighting value. For example, we could configure queue 5 and queue 4 as members of the WRR group. We configure queue 5 with a weighting of 10, and queue 4 with a weighting of 20. When the egress interface is oversubscribed, two frames will egress queue 4 for every packet that egresses queue 5.

Pause frames and their limitations

The traditional solution to over-subscription has been Pause Frames. The IEEE 802.3x standard introduced these as a basic flow control mechanism for Ethernet. When a device becomes congested, it sends a pause frame to its peer, instructing it to stop transmitting for a specified duration. This prevents buffer overflow and packet loss.

However, pause frames apply to the entire link. All traffic, regardless of type or importance, is paused. This can lead to performance degradation, especially in networks carrying multiple traffic classes. For example, pausing voice or video traffic due to congestion in a low-priority data stream can cause noticeable service disruption.

Priority-based flow control

To improve on simple pause frames, PFC enhances traditional flow control by introducing per-priority pause capability. Instead of pausing all traffic, a device can pause only the traffic associated with a specific priority level.

This is achieved by mapping traffic to different traffic classes and configuring devices to respond to congestion on a per-priority, per-class basis.

PFC uses a modified pause frame format that includes a bitmap indicating which priorities should be paused. This allows devices to maintain throughput for high-priority traffic while controlling congestion in lower-priority queues.

There are a number of benefits of PFC:

- lossless Ethernet for critical traffic: PFC enables lossless transmission for selected traffic classes, which is essential for storage, AI, and high performance compute (HPC) applications.
- improved network efficiency: By pausing only congested traffic, PFC helps maintain overall link utilization and reduces unnecessary delays.
- granular traffic control: Network administrators can define policies that prioritize traffic based on business needs, ensuring that critical services are not impacted by congestion.
- Compatibility with Data Center Bridging (DCB): PFC is a key component of DCB, making it suitable for modern data center environments.

PFC is particularly useful in environments where Ethernet is used for storage transport (e.g., iSCSI, RoCE), high-performance computing, or HCI environments. It is also beneficial in networks that implement Quality of Service (QoS) policies and require differentiated handling of traffic types.

Set the CoS-to-queue mappings if PFC is used on queues 0, 1 and 2

By default, most of the queues map directly to CoS values of the same number (e.g. queue 7 maps to CoS 7). This is not the case for queues 0, 1, and 2. If you are using PFC on those queues, we recommend you change their mapping.

For example, imagine if PFC is enabled on all queues on a given port. If queue **1** becomes oversubscribed, and queue **1** is mapped to CoS value **0** as in the default settings, then a PFC pause frame will be sent. This frame means that the rate of traffic being sent with CoS **0** should be reduced. However, you may expect that pausing queue 1 would instead pause traffic with CoS 1.

To avoid this, it is generally recommended symmetric CoS-to-queue mappings be used when using PFC (CoS value 2 to queue 2, CoS value 3 to queue 3, etc.).

By default, the CoS-to-queue mappings on AlliedWare Plus devices are as follows:

```
COS-TO-QUEUE-MAP:
COS   :   0   1   2   3   4   5   6   7
-------------------------------------
QUEUE:    2   0   1   3   4   5   6   7
```

So, if you use PFC on queues 0,1, or 2 we recommend that you remap the CoS-to-queue values. See “Example: configuring PFC on an interface” for an example of this procedure.

Lossless egress pool

For PFC to operate in a lossless manner a lossless egress pool must be configured, and all PFC enabled priorities assigned to the lossless pool.

When you have configured the lossless pool, a percentage of the total pool is dedicated to the lossless traffic classes. The remainder of the egress pool is available to the remaining lossy traffic classes. This means that if the lossy traffic consumes its portion of the egress pool, it will not affect the designated lossless traffic classes.

Data Center Bridging eXchange protocol (DCBX)

DCBX is used by Data Center Bridging (DCB) devices to exchange configuration information with directly connected peers. The protocol may also be used for misconfiguration detection and for configuration of the peer.

DCBX uses LLDP to exchange attributes between two links peers. When DCBX and LLDP are enabled on an interface, the following Type-Length-Value (TLV) elements will be advertised:

Table 1-1: ETS Configuration TLV — D.2.9 of IEEE Std 802.1Q-2018

Field	Description	AlliedWare Plus Support
Willing	Indicates if the device is willing to accept configuration from neighbors	Always 0 to indicate unwilling
Credit Based Shaper	Indicates if the device supports the Credit-based Shaper transmission selection algorithm	Always 0 to indicate unsupported
Max Traffic Classes	Indicates the maximum number of traffic classes the device supports.	Always 0 to indicate support for 8 traffic classes
Priority Assignment Table	Mapping of priority to traffic classes	Mappings as configured by mls qos map cos-queue <0-7> to <0-7>
Traffic Class Bandwidth Table	Indicates the current bandwidth percentage configured for each traffic class	Percentages as configured by mls qos scheduler-set <1-12> wrr-queue group 1 percent <1-100> queue <0-7>
TSA Assignment Table	Indicates the Transmission selection algorithm to be used for each traffic class	ETS (2) for traffic classes configured with wrr-queue and percent. Otherwise Strict Priority (0)

Additionally, if PFC is enabled on an interface, the following TLV elements will be advertised:

Table 1-2: PFC TLV — D.2.11 of IEEE Std 802.1Q-2018

Item	Default profile	Profile1
Willing	Indicates if the device is willing to accept configuration from neighbors	Always 0 to indicate unwilling
MACsec Bypass Capability	Indicates if the device is capable of bypassing MACsec processing when MACsec is disabled	Always 0 to indicate capable
PFC Capability	Indicates the maximum number of traffic classes that simultaneously support PFC on the device	Always 8
PFC Enable	Indicates if PFC is enabled for each the priority	As configured by pfc priority <0-7>

Configuration examples

In this section, you can find examples of how to configure the features described above.

Example: configuring PFC on an interface

In the following example, you want to enable PFC for priorities 2 and 3 on port1.0.1.

You can configure this example with the following process.

Step 1: **Enter configuration mode**

Enter configuration mode for the device. Use the command:

```
awplus#configure terminal
```

Step 2: **Enable the PFC service**

Enable the PFC service on the device. Use the command:

```
awplus(config)#service pfc
```

Step 3: **Enter interface configuration mode**

Enter interface configuration mode for port1.0.1. Use the command:

```
awplus(config)#interface port1.0.1
```

Step 4: **Enable PFC mode**

Enable PFC mode for the interface. Use the command:

```
awplus(config-if)#pfc mode on
```

Note: This command has options of on, auto and off. The default is **auto**. To enable PFC on an interface, set the mode to **on**. Auto mode is not supported in AlliedWare Plus version 5.5.5-2, and behaves the same as off in this version. In a future release, the auto mode will allow for auto-negotiation of PFC settings using DCBX.

Step 5: **Configure the PFC priorities**

The priority settings specify which of the interface's egress queues are managed by PFC, so to set it, use the **pfc priority** command with the desired queue number. You can specify multiple queues by entering this command multiple times.

Configure the PFC priorities for the interface. Use the commands:

```
awplus(config-if)#pfc priority 2
awplus(config-if)#pfc priority 3
```

So now, in our example, PFC is enabled for priority 2 and 3 on port1.0.1.

Step 6: **Return to global configuration mode and remap the COS-to-queue mappings**

Because PFC is being used on queue 2 in this example, we recommend you remap the CoS-to-queue values for queues 0, 1 and 2. Use the following commands:

```
awplus(config-if)#exit
awplus(config)#mls qos map cos-queue 0 to 0
awplus(config)#mls qos map cos-queue 1 to 1
awplus(config)#mls qos map cos-queue 2 to 2
```

Step 7: **Check the config**

You can confirm that PFC is configured by viewing the config with the **show running-config** command:

```
awplus# show run
...
!
service pfc
!
...
!
interface port1.0.1
switchport
switchport mode access
pfc mode on
pfc priority 2
pfc priority 3
...
!
mls qos map cos-queue 0 to 0
mls qos map cos-queue 1 to 1
mls qos map cos-queue 2 to 2
!
```

To view information about buffer information relevant to PFC, use the following command:

```
awplus# show platform mem QosBufferConfig
```

Example: configuring the lossless egress pool

In the following example, you have completed the previous example, and enabled PFC on port1.0.1. Now, you want to configure the lossless egress pool. You want to add the two PFC enabled priorities, 2 and 3, and allocate a buffer percentage of 10%. This will allow the device to buffer egress data for those priorities.

You can configure this example with the following process.

Step 1: **Enter configuration mode**

Enter configuration mode for the device. Use the command:

```
awplus#configure terminal
```

Step 2: **Enable the lossless egress pool**

Enable the lossless egress pool on the device. This will also put the device in **config-qos-egress** mode. Use the command:

```
awplus(config)#mls qos egress-pool lossless
```

Step 3: **Assign the priorities**

Assign the PFC-enabled priorities to the lossless pool. Use the commands:

```
awplus(config-qos-egress)#priority 2
awplus(config-qos-egress)#priority 3
```

Step 4: **Set the buffer percentage**

Set the buffer percentage for the lossless pool. Use the command:

```
awplus(config-qos-egress)#buffer-percentage 10
```

Step 5: **Return to privileged exec mode**

Return to privileged exec mode for the device. Use the command:

```
awplus(config-qos-egress)#end
```

Step 6: **Update the config file**

Update the config file by writing your changes to the startup config file. Use the command:

```
awplus#write
```

Step 7: **Reboot the device**

Reboot the device to apply the changes. Use the command:

```
awplus#reboot
```

So now, in our example, the lossless egress pool has been enabled and configured, and the PFC-enabled priorities, 2 and 3, have been added.

Example: configuring DCBX on an interface

In the following example, you have completed the previous examples, enabled PFC on port1.0.1, and configured the lossless egress pool. Now, you want to enable DCBX on port1.0.1. This will allow the interface to advertise its PFC settings.

You can configure this example with the following process.

Step 1: **Enter configuration mode**

Enter configuration mode for the device. Use the command:

```
awplus#configure terminal
```

Step 2: **Enable the DCBX service**

Enable the DCBX service on the device. Use the command:

```
awplus(config)#service dcbx
```

Step 3: **Enter interface configuration mode**

Enter interface configuration mode for port1.0.1. Use the command:

```
awplus(config)#interface port1.0.1
```

Step 4: **Enable DCBX for PFC**

Enable DCBX for PFC on the interface. Use the command:

```
awplus(config-if)#dcbx pfc
```

Step 5: **Return to global configuration mode**

Return to global configuration mode for the device. Use the command:

```
awplus(config-if)#exit
```

Step 6: **Enable LLDP**

Enable LLDP for the interface. Use the command:

```
awplus(config)#lldp run
```

So now, in our example, port1.0.1 is using DCBX to advertise its PFC settings.

Step 7: **Check LLDP settings**

You can view the currently advertised PFC parameters with the **show lldp local-info** command:

```
awplus#show lldp local-info dot1 interface port1.0.1

LLDP Local Information:

Local port1.0.1:
  Chassis ID Type ................... MAC address
  Chassis ID ....................... 889d.98df.23b2
  Port ID Type ..................... Interface name
  Port ID .......................... port1.0.1
  TTL .............................. 120
  Port VLAN ID (PVID) .............. 1
  Port & Protocol VLAN - Supported . Yes
                       - Enabled ... No
                       - VIDs ..... 0
  VLAN Names ....................... default
  Protocol IDs ..................... 9000, 0026424203000000, 0027424203000002,
                                     0069424203000003, 888e01,
                                     aaaa0300e02b00bb, 88090101, 00540000e302,
                                     000a424203000101, 0800, 0806, 86dd,
                                     89020001, 89020128
  ETS Configuration:
       Willing ..................... 0
       CBS ......................... 0
       Max TCs ..................... 0
       Priority Assignment Table ... [0, 1, 2, 3, 4, 5, 6, 7]
       TC Bandwidth Table .......... [0, 0, 10, 10, 0, 0, 0, 0]
       TSA Assignment Table ........ [0, 2, 0, 2, 0, 2, 0, 2]
  ETS Recommendation:
       Priority Assignment Table ... [0, 0, 0, 0, 0, 0, 0, 0]
       TC Bandwidth Table .......... [0, 0, 0, 0, 0, 0, 0, 0]
       TSA Assignment Table ........ [0, 0, 0, 0, 0, 0, 0, 0]
  PFC:
       Willing ..................... 0
       MBC ......................... 0
       PFC Capability............... 8
       PFC Enable................... 4
```

This example is using the **dot1** parameter to show 802.1 TLVs, and the **interface** parameter to show the details for port1.0.1.

You can view the PFC parameters received from neighbors with the **show lldp neighbors detail** command:

```
awplus#show lldp neighbors detail dot1 interface port1.0.1

LLDP Detailed Neighbor Information:

Local port1.0.1:
  Neighbors table last updated 0 hrs 0 mins 2 secs ago

  Chassis ID Type ................... MAC address
  Chassis ID ........................ 0000.cd37.0005
  Port ID Type ...................... Interface name
  Port ID ........................... port1.0.1
  TTL ............................... 120 (secs)
  Port VLAN ID (PVID) ............... [not advertised]
  Port & Protocol VLAN .............. [not advertised]
  VLAN Names ........................ [not advertised]
  Protocol IDs ...................... [not advertised]
  ETS Configuration:
       Willing ...................... 0
       CBS .......................... 0
       Max TCs ...................... 8
       Priority Assignment Table ... [0, 1, 2, 3, 4, 5, 6, 7]
       TC Bandwidth Table ........... [0, 0, 10, 10, 0, 0, 0, 0]
       TSA Assignment Table ......... [0, 0, 0, 2, 0, 2, 0, 0]
  ETS Recommendation ................ [not advertised]
  PFC:
       Willing ...................... 0
       MBC .......................... 0
       PFC Capability................ 8
       PFC Enable.................... 3, 5
  Application Priority .............. TCP/SCTP port 3260
       Priority ..................... 5
  Application VLAN .................. [not advertised]
```

This example is using the **dot1** parameter to show 802.1 TLVs, and the **interface** parameter to show the details for port1.0.1.