



White Paper

FPIN Verification



Fabric Performance Impact Notification (FPIN)

Version 4.0

Author

Gairy Grannum
Sr. Principal Software Engineer
SANBlaze

Table of Contents

- Fabric Network Performance Notification (FPIN)..... 1
- Configuring FPIN in the SANBlaze Software 1
 - FPIN Setting..... 2
 - FPIN Disable Notifications..... 2
 - FPIN Disable Logging Received Notifications..... 3
 - FPIN Disable Logging Sent Notifications 3
- Reporting FPIN Information in the SANBlaze Software..... 3
 - Virtual Port FPIN Displays 3
 - FPIN Counters 5
 - FPIN System Logging..... 6
- Sending FPINs through the SANBlaze Software..... 8
 - FPINs Sent Through the SANBlaze GUI 8
 - FPINs Sent Through the SANBlaze Command Line Interface (CLI) 10
 - VPI 10
 - FCID 10
 - Burst 10
 - Descriptors 10
 - Link Integrity Event 11
 - Examples of sending FPIN with Link Integrity descriptor: 11
 - Delivery Event 11
 - Examples of sending FPIN with Delivery descriptor 12
 - Peer Congestion Event..... 12
 - Example of sending FPIN with Peer Congestion descriptor 12
 - Congestion Event 12
 - Example of sending FPIN with Peer Congestion descriptor 13
- Conclusion..... 13

Fabric Network Performance Notification (FPIN)

Network congestion is a problem in data networks that can impact the performance of the network. Identifying the source of the congestion and mitigating it can be a difficult task.

To help address network congestion, FPINs are defined to help identify types and sources of congestion and allow the receivers of the information to act in a manner to mitigate the congestion.

The notifications are delivered through FPIN Extended Link Service (ELS) and a new congestion primitive. Determination of whether the congestion primitive is supported is determined through the negotiation with the Exchange Diagnostic Capabilities (EDC) ELS. Determination of which types of notifications are supported by the switch is negotiated with the Register Diagnostic Functions (RDF) ELS. The negotiated types of notification are:

- Link Integrity notification
- Delivery notification
- Peer Congestion notification
- Congestion notification (if not negotiated with the EDC ELS)

Switch vendors recommend you enable FPIN when you first log into the switch, but theoretically, you can enable FPIN anytime.

To allow the debugging of sent and received FPINs, the SANBlaze solution provides mechanisms to allow FPINs with various notifications in the payload to be sent to the Fabric Controller and specified N_PORTS. The solution also allows for the receiving of FPINs. Logging and tracing is provided for any FPIN received or sent. In addition, counters are provided for the notifications contained within the FPINs which are sent or received. The SANBlaze solution also allows viewing of the notifications contained within the FPINs sent and received.

Configuring FPIN in the SANBlaze Software

Configuring the FPIN feature is done through the configuration page of the port. In the main page of the SBExpress GUI, select the Fabric port for which you would like to configure FPIN. Once you select the desired port, select the Configuration tab.

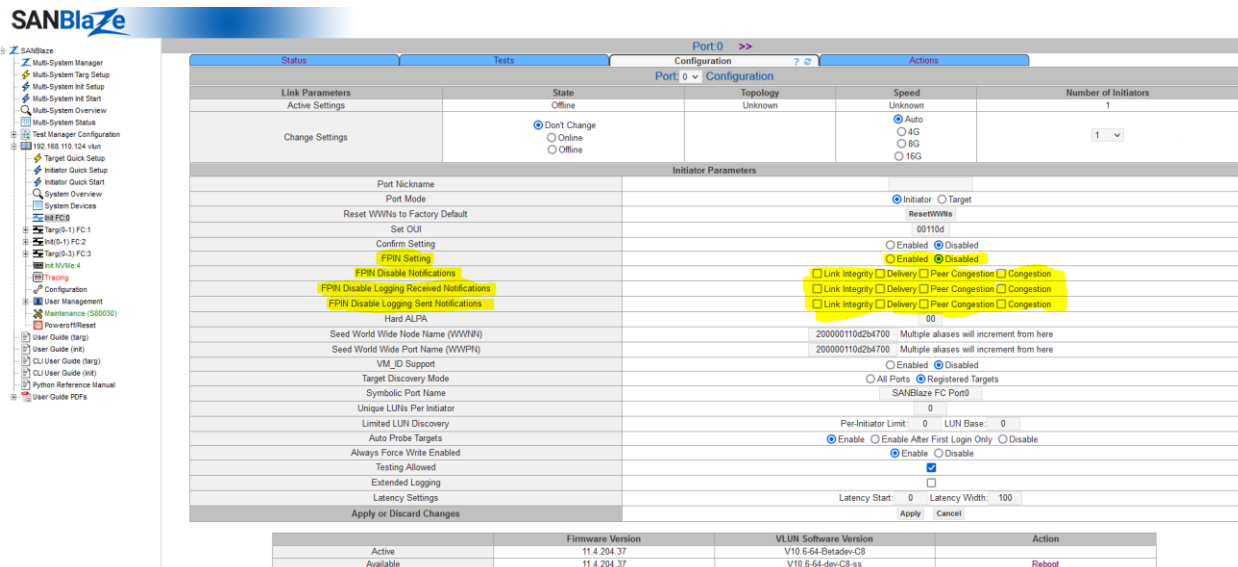


Figure 1: FPIN Setting on Configuration Page

FPIN Setting

- **Disabled** – FPIN receiving is disabled on this port. This means negotiation of FPIN support will not occur with the switch. In this case, the Fabric Controller should ignore any FPINs sent to it by the port and the port will ignore any FPINs received. However, to allow for testing of FPINs being received by the switch or other N_Ports, sending of FPINs through the port is still allowed.
- **Enabled** – FPIN receiving is enabled. The port will negotiate with the switch which link descriptors are supported.

NOTE: You can hover over each parameter for a description of the setting.

FPIN Disable Notifications

- **Link Integrity** – If checked, disables processing of the link integrity descriptor when an FPIN is received. This will also trigger a negotiation with the switch which will exclude this descriptor. The unexpected count for the Receive Link Integrity counter will be incremented to show an FPIN was received with this descriptor.
- **Delivery** – If checked, disables processing of the delivery descriptor when an FPIN is received. This will also trigger a negotiation with the switch which will exclude this descriptor. The unexpected count for the Delivery counter will be incremented to show an FPIN was received with this descriptor.
- **Peer Congestion**– If checked, disables processing of the Peer Congestion descriptor when an FPIN is received. This will also trigger a negotiation with the switch which will exclude this descriptor. The unexpected count for the Peer Congestion counter will be incremented to show an FPIN was received with this descriptor.
- **Congestion**– If checked, disables processing of the Congestion descriptor when an FPIN is received. This will also trigger a negotiation with the switch which will exclude this descriptor. The unexpected count for the Congestion counter will be incremented to show an FPIN was received with this descriptor.

NOTE: You can hover over each parameter for a description of the setting.

FPIN Disable Logging Received Notifications

- **Link Integrity** – If checked, disables logging of Link Integrity descriptors received to the Linux message system (/var/log/messages). This prevents excessive logging when testing receiving FPINs with Link Integrity descriptors.
- **Delivery** – If checked, disables logging of Delivery descriptors received to the Linux message system (/var/log/messages). This prevents excessive logging when testing receiving FPINs with Delivery descriptors.
- **Peer Congestion**– If checked, disables logging of Peer Congestion descriptors received to the Linux message system (/var/log/messages). This prevents excessive logging when testing receiving FPINs with Peer Congestion descriptors.
- **Congestion**– If checked, disables logging of Congestion descriptors received to the Linux message system (/var/log/messages). This prevents excessive logging when testing receiving FPINs with Congestion descriptors.

NOTE: You can hover over each one for a description of the setting.

FPIN Disable Logging Sent Notifications

- **Link Integrity** – If checked, disables logging of Link Integrity descriptors sent to the Linux message system (/var/log/messages). This prevents excessive logging when testing sending FPINs with Link Integrity descriptors.
- **Delivery** – If checked, disables logging of Delivery descriptors sent to the Linux message system (/var/log/messages). This prevents excessive logging when testing sending FPINs with Delivery descriptors.
- **Peer Congestion**– If checked, disables logging of Peer Congestion descriptors sent to the Linux message system (/var/log/messages). This prevents excessive logging when testing sending FPINs with Peer Congestion descriptors.
- **Congestion**– If checked, disables logging of Congestion descriptors sent to the Linux message system (/var/log/messages). This prevents excessive logging when testing sending FPINs with Congestion descriptors.

NOTE: You can hover over each parameter for a description of the setting.

Reporting FPIN Information in the SANBlaze Software

Virtual Port FPIN Displays

When FPIN is enabled on the port, two new rows of information are added to the virtual port's configuration display.

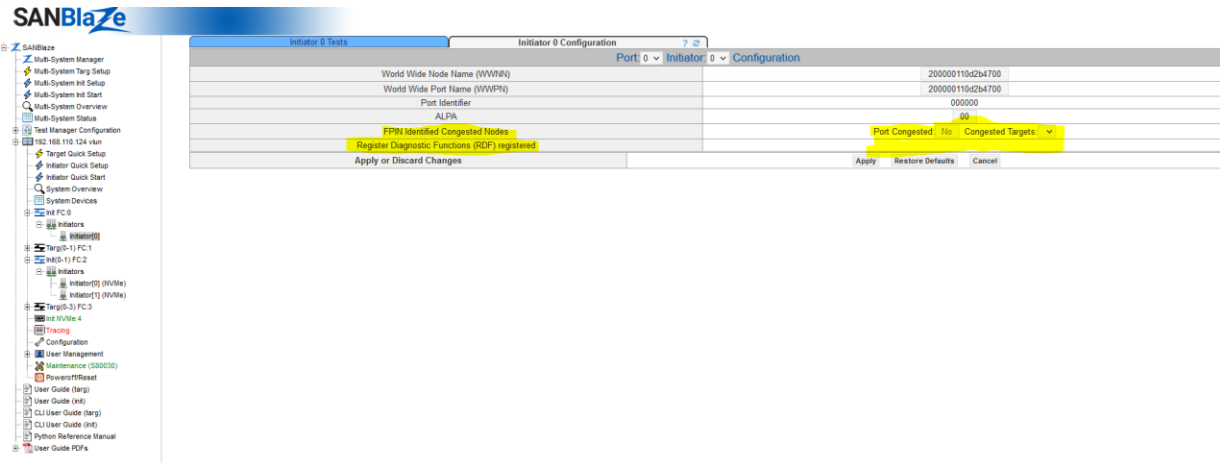


Figure 2: Virtual Port with FPIN Displays

- FPIN Identified Congested Nodes** – shows the nodes which are zoned and communicates with this port that has been reported to have congestion (Peer Congestion notification). An initiator port shows the targets which it is logged into that have congestion under the “Congested Targets” drop down list. A target port shows the initiators which it is logged into that have congestion under the “Congested Initiators” drop down list. If the virtual port has been reported to have congestion, the Port Congested dialog box will be set to Yes. Otherwise, the display shows No.
- Register Diagnostic Functions (RDF) registered** – shows which FPIN descriptors have been registered with the switch through the RDF ELS message for this virtual port.

The following is an example of the display when the port is online, no FPIN descriptors have been disabled, and the RDF exchange was successful.

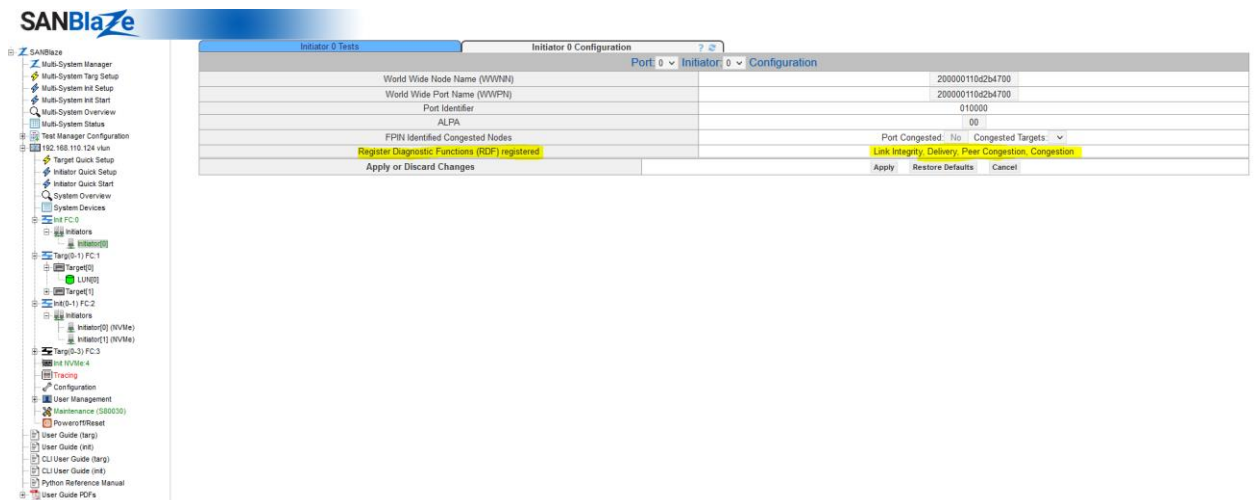


Figure 3: Virtual Port FPIN Descriptors Registered with Switch

The following is an example of the display when the port is online, no FPIN descriptors have been disabled, the RDF exchange was successful, and congestion has been reported for one of the initiator's targets.

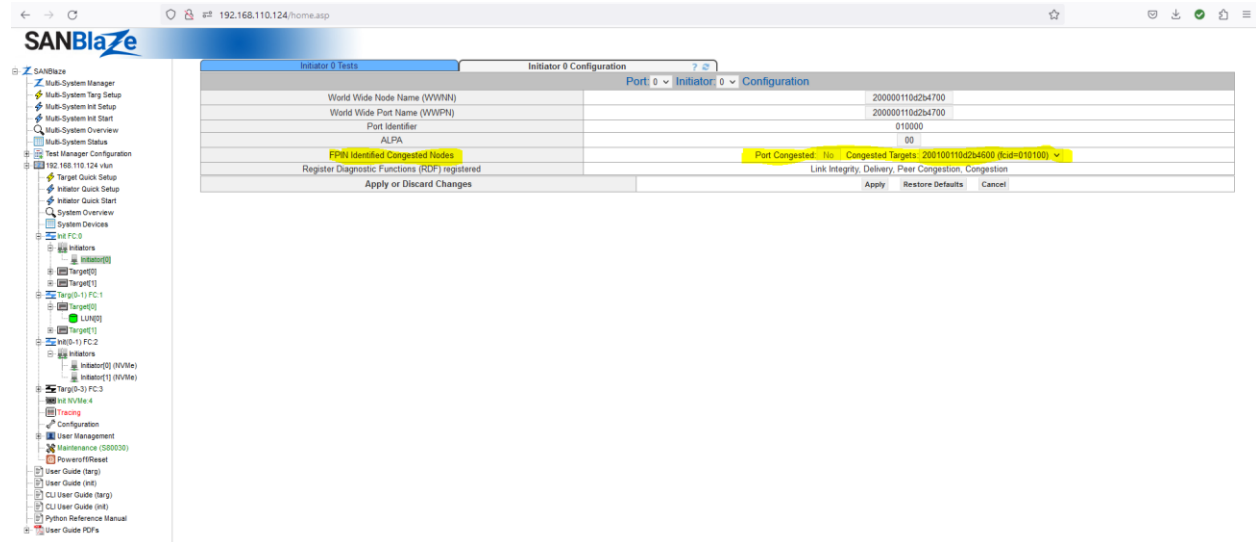


Figure 4: Virtual Port with Congestion Reported

FPIN Counters

On the status page of the port, there is a section for Received FPINs counters and Sent FPINs counters.

For each possible descriptor that can be sent or received there is a row representing that descriptor. Each descriptor has a counter for the support event types plus two additional counters. The “Other” counter represents a descriptor whose event type is not known by the SANBlaze software. The “Unexpected” counter represents receiving the descriptor when it’s unexpected. This currently means the descriptor was disabled. Each descriptor has a hyperlink that allows showing the specifics of the descriptors that have been received or sent.

The last row reports the total number of FPINs received for the Received FPINs or sent for the Sent FPINs. There is a hyperlink that allows showing the specifics of all descriptors received or sent.

The following are examples of an initiator port that received 119 Peer Congestion Notifications and a clear event.

FLOGI/FDISC Statistics		Sent: 1	Acc'd: 1	Rejected: 0	Clear Statistics
Received FPIs					
Link Integrity	Unknown: 0	Link Failure: 0	Loss-of-Synchronization: 0	Loss-of-Signal: 0	
	Primitive Sequence Protocol Error: 0	Invalid Transmission Word: 0	Invalid CRC: 0	Device Specific: 0	
Delivery	Other: 0	Unexpected: 0			
	Unknown: 0	Timeout: 0	Unable to route: 0	Device Specific: 0	
Peer Congestion	Other: 0	Unexpected: 0			
	Lost Credit: 119	Credit Stall: 0	Oversubscription: 0	Device Specific: 0	
Congestion	Cleared: 1	Other: 0	Unexpected: 0		
	Warning Lost Credit: 0	Warning Credit Stall: 0	Warning Oversubscription: 0	Warning Device Specific: 0	
	Warning Cleared: 0	Warning Other: 0	Warning Unexpected: 0		
	Alarm Lost Credit: 0	Alarm Credit Stall: 0	Alarm Oversubscription: 0	Alarm Device Specific: 0	
	Alarm Cleared: 0	Alarm Other: 0	Alarm Unexpected: 0		
	OtherSeverity Lost Credit: 0	OtherSeverity Credit Stall: 0	OtherSeverity Oversubscription: 0	OtherSeverity Device Specific: 0	
OtherSeverity Cleared: 0	OtherSeverity Other: 0	OtherSeverity Unexpected: 0			
All Received FPIs	Total: 120			Clear Recv FPIs	
Sent FPIs					
Link Integrity	Unknown: 0	Link Failure: 0	Loss-of-Synchronization: 0	Loss-of-Signal: 0	
	Primitive Sequence Protocol Error: 0	Invalid Transmission Word: 0	Invalid CRC: 0	Device Specific: 0	
Delivery	Other: 0	Timeout: 0	Unable to route: 0	Device Specific: 0	
	Unknown: 0				
Peer Congestion	Other: 0	Unexpected: 0			
	Lost Credit: 0	Credit Stall: 0	Oversubscription: 0	Device Specific: 0	
Congestion	Cleared: 0	Other: 0			
	Warning Lost Credit: 0	Warning Credit Stall: 0	Warning Oversubscription: 0	Warning Device Specific: 0	
	Warning Cleared: 0	Warning Other: 0	Warning Unexpected: 0		
	Alarm Lost Credit: 0	Alarm Credit Stall: 0	Alarm Oversubscription: 0	Alarm Device Specific: 0	
	Alarm Cleared: 0	Alarm Other: 0	Alarm Unexpected: 0		
	OtherSeverity Lost Credit: 0	OtherSeverity Credit Stall: 0	OtherSeverity Oversubscription: 0	OtherSeverity Device Specific: 0	
OtherSeverity Cleared: 0	OtherSeverity Other: 0	OtherSeverity Unexpected: 0			
All Sent FPIs	Total: 0			Clear Sent FPIs	

Figure 5: Received FPI Counters

Information displayed when the Peer Congestion hyperlink is enabled.

The screenshot shows the SANBlaze management interface. On the left is a navigation tree with categories like 'SANBlaze', 'Multi-System Manager', 'Initiator', 'Target', and 'Configuration'. The main area displays 'FLOGI/FDISC Statistics' for 'Sent: 0'. A 'Peer Congestion' link is highlighted in yellow. A modal window titled 'SANBlaze Virtual LUN Port Detail' is open, showing detailed information for 'SANBlaze Virtual LUN Port 2 Detail'. The modal lists 'PeerCongestion' events with details such as 'Lost Credit: Detecting Portname 0x200300110ddb6b00, Attached Portname 0x200200110ddb6c00, Event Threshold 0x400, Portname list - 0x200300110ddb6b00 Received 119 times', 'Received FPI on Dec: 18 13:04:28.941038 from fcid 010200', and 'Clear/None: Detecting Portname 0x200300110ddb6b00, Attached Portname 0x200200110ddb6c00, Event Threshold 0x400, Portname list - 0x200300110ddb6b00 Received FPI on Dec: 18 13:06:37.503993 from fcid 010200'.

FPI System Logging

If FPI is enabled and logging of the FPI descriptor is not disabled by the “FPI Disable Logging Receive Notifications” setting on the configuration page, FPI descriptors received will be logged to the `/var/log/messages` file.

In the case where the initiator port received 119 Peer Congestion Notifications and a clear event, the logs look like:

```
“Dec 18 13:04:28 vln kernel: [ 9444.947472] Received FPI PEER CONGESTION NOTIFICATION: d_id 010200 detecting portname 0x200300110ddb6b00, attached
```


portname 0x200200110ddb6c00, event type Lost Credit, event threshold 0400,
portnames: 200300110ddb6b00

Dec 18 13:06:37 vlun kernel: [9574.413458] Received FPIN PEER CONGESTION
NOTIFICATION: d_id 010200 detecting portname 0x200300110ddb6b00, attached
portname 0x200200110ddb6c00, event type Lost Credit, event threshold 0400,
portnames: 200300110ddb6b00, received 118 additional times

Dec 18 13:06:37 vlun kernel: [9574.413470] Received FPIN PEER CONGESTION
NOTIFICATION: d_id 010200 detecting portname 0x200300110ddb6b00, attached
portname 0x200200110ddb6c00, event type Clear/None, event threshold 0400,
portnames: 200300110ddb6b00"

If FPIN is enabled and logging of the FPIN descriptor is not disabled by the
"FPIN Disable Logging Sent Notifications" setting on the configuration page,
FPIN descriptors sent will be logged to the /var/log/messages file.

In the case where the initiator port received 119 Peer Congestion
Notifications and a clear event, the sender logs look like:

"Dec 18 13:04:28 vlun kernel: [9444.947401] Sending FPIN ELS from 0x10300 to
0xfffffd a total of 120 times at 1024 millisecond intervals

Dec 18 13:06:37 vlun kernel: [9574.413418] FPIN ELS from 0x10300 to 0x10200
was sent 120 times

Dec 18 13:06:37 vlun kernel: [9574.413425] Sent FPIN PEER CONGESTION
NOTIFICATION: s_id 010300, d_id 010200 detecting portname 0x200300110ddb6b00,
attached portname 0x200200110ddb6c00, event type Lost Credit, event threshold
0400, portnames: 200300110ddb6b00, sent 119 times followed by a clear event"

Sending FPINs through the SANBlaze Software

FPINs can be sent from both the SANBlaze GUI and the Command Line Interface (CLI). In both cases, the FPIN may be sent once or for a duration at a specified interval.

FPINs Sent Through the SANBlaze GUI

In the **Actions** menu, scroll down until you see the FC Send FPINs section.

FC Send FPINs		Select
Notification		
Destination/Source	<input checked="" type="radio"/> Send to Fabric <input type="radio"/> Send to N_PORT Id <input type="radio"/> Send from Target 0	
Attached Portname	Portname: 200200110ddb6c00 (fcid=010200) Specified Portname: 0000000000000000	
Portname List	<input type="radio"/> No List <input checked="" type="radio"/> List <input type="radio"/> Specify List Specified portnames list:	
Send Burst	Send Interval: 0 Send Count: 0 Send Clear Event <input type="checkbox"/>	<input type="checkbox"/>
Link Integrity	Event Type: 0 Event Modifier: 0 Event Threshold: 0 Event Count: 0	<input type="checkbox"/>
Delivery	Reason Code: 0 Event Data:	<input type="checkbox"/>
Peer Congestion	Event Type: 0 Event Modifier: 0 Event Threshold: 0	<input type="checkbox"/>
Congestion	Event Type: 0 Event Modifier: 0 Event Threshold: 0 Severity: 0	<input type="checkbox"/>
		Send FPIN

Figure 2: FC Send FPINs parameters

- **Destination / Source** contains the following choices:
 - **Send to Fabric** – Send FPIN to Fabric.
 - **Send to N_PORT Id:** – Send FPIN to N_PORT by entering the port ID.
 - **Send from Target/Initiator** – Choose the Target (if target port) or Initiator (if initiator port) to send the FPIN from.
- **Attached Portname** (see FC-LS-5 for definition):
 - **Portname** – Drop down list of the available nodes that can be specified as the attached portname for the link integrity, delivery, or peer congestion descriptors. The list of wwpns/fcid displayed will correspond to the nodes the target or initiator is zoned and logged into. The last option in the drop down is “Specify Portname” which allows the Specified Portname field to be used.
 - **Specified Portname** – Allows the portname to be specified. This allows the sending node to be specified as the attached portname, another member of the zone, or a node outside the zone.
- **Portname List** (see FC-LS-5 for definition) – Specifies the portname list for the link integrity and/or peer congestion notifications. If the FPINs are sent to the switch with the intention of the switch processing the FPINs, a **LIST MUST BE SPECIFIED**.
 - **No List** – Portname list not included for link integrity and/or peer congestion notifications.
 - **List** – The SANBlaze software generates the appropriate list for the descriptor. This is the default setting.
 - **Specify List** – Allows the portname to be specified. The portnames are hex values specified by commas.
- **Send Burst** – if the check box is enabled, allows the FPIN to be sent a number of times at a specified interval. If the Peer Congestion or Congestion descriptor is specified, allows a clear event to be sent in the final FPIN.
 - **Send Interval** – A decimal value specifying the number of milliseconds between sending FPINs. If 0, a delay will not occur when sending the FPIN notifications.
 - **Send Count** – A decimal value specifying the number of times the FPIN will be sent. A minimum of 2 is required.

- **Send Clear Event** – If a Peer Congestion or Congestion descriptor is being sent and the check box is checked, the last FPIN sent will have a clear event for the Peer Congestion or Congestion descriptor.
- **Link Integrity** (see FC-LS-5 for definition) – The link integrity field provides the following parameters:
 - **Event Type** – Type of Link Integrity event in hex.
 - **Event Modifier** – Implement specific value when event type is device-specific in hex. This is normally zero for other event types.
 - **Event Threshold** – Duration in milliseconds of link integrity detection cycle in hex.
 - **Event Count** – Minimum number of event occurrences during the Event Threshold to cause generation of the Link Integrity Event in hex.
- **Delivery** (see FC-LS-5 for definition) – The following are the delivery parameters:
 - **Reason Code** – Reason for the delivery failure in hex.
 - **Event Data** – The 24-byte frame header of the discard frame. Separate each hex dword with commas.
- **Peer Congestion** (see FC-LS-5 for definition) – The following are the peer congestion parameters:
 - **Event Type** – Type of peer congestion event in hex.
 - **Event Modifier** – Implement specific value when event type is device specific in hex. This is normally zero for other event types.
 - **Event Threshold** – Duration in milliseconds of the detected peer congestion event in hex. If sent using the Send Burst option, this value must match the Send Interval value of the Send Burst option.
- **Congestion** (see FC-LS-5 for definition) – The following are the parameters for congestion:
 - **Event Type** – Type of congestion event in hex.
 - **Event Modifier** – Implement specific value when event type is device specific in hex. This is normally zero for other event types.
 - **Event Threshold** – Duration in milliseconds of the detected congestion event in hex. If sent using the Send Burst option, this value must match the Send Interval value of the Send Burst option.
 - **Severity** – Status contains a value indicating the urgency level of the notification in hex.
- **Checkboxes** – The checkboxes under the Select column next to Link Integrity, Delivery, Peer Congestion, and Congestion signify to include those when sending the FPIN. The checkbox next to Send Burst enables the sending of the FPINs for a duration. The following are options for each:
 - Send FPINs the number of times specified
 - Include **Link Integrity** in FPIN
 - Include **Delivery** in FPIN
 - Include **Peer Congestion** in FPIN
 - Include **Congestion** in FPIN
- **Send FPIN** – Sends the FPIN with the included specified parameters.

Figure 3: Detailed Status

FPINs Sent Through the SANBlaze Command Line Interface (CLI)

An FPIN ELS can be sent through the CLI with the **SendFpin** command. The general format of the **SendFpin** command is **SendFpin=Vpi,FCID,[Burst,]Descriptors**. The command can be directed to the target or initiator port from which the FPIN ELS is sent to.

VPI

The VPI is the ID of the virtual initiator or virtual target on the port from which the ELS is sent. For example, if the port has 5 virtual initiators and the ELS is to be sent from initiator[1], the vpi will be 1.

FCID

The FCID is the hexadecimal 24-bit FC ID assigned to the target or initiator that the ELS will be sent to.

Burst

Burst is an optional field that is only specified if more than one FPINs are to be sent with this command. The format of Burst is:

Burst=interval:cnt:clear

Interval – hex value of the number of milliseconds to wait before sending the next FPIN

Cnt – hex value of the number of FPINs to send.

Clear – non zero decimal value indicates if Peer Congestion or Congestion descriptors are included, the last FPIN sent will be a clear event for those descriptors.

Descriptors

The descriptors define the notifications to be included in the FPIN ELS. One or more notifications may be included in FPIN ELS, so one or more descriptors can be specified. The start of the descriptor is identified by the key word “Event=” and terminated with a semi-colon (“;”). The following sections describe the events that correspond to the notifications.

Link Integrity Event

The format of the Link Integrity event is:

LinkIntegrity:APortName,EventType,EventModifier,EventThreshold,EventCount[,PortNameList];

APortName – 64-bit hex value of the attached Port Name.

EventType – hex value of the event type.

EventModifier – hex value of the event modifier.

EventThreshold – hex value of the event threshold.

EventCount – hex value of the event count.

PortNameList – if the default portname list desired, a value of 0 is specified. If the port name list is not to be included, the ‘;’ terminates the event after the EventCount is specified. Otherwise, a comma separated list of 64-bit hex values representing the port names.

Examples of sending FPIN with Link Integrity descriptor:

Sending the descriptor to fc id 0x010000 from vpi 0 of port1 with:

APortName – 200000110d2b4700

EventType – 1 (Link Failure).

EventModifier – 0

EventThreshold – 100

EventCount – 0

PortNameList – 0

SendFpin=0,010000,Event=LinkIntegrity:200000110d2b4700,1,0,100,4,0; > /port1/port

Sending the same descriptor to the fabric controller:

SendFpin=0,010000,Event=LinkIntegrity:200000110d2b4700,1,0,100,4,0; > /port1/port

Sending the same descriptor 3 times at 1024 second intervals (the intervals for the burst and the descriptor do not need to match):

SendFpin=0,010000,Burst=400:3:0,Event=LinkIntegrity:200000110d2b4700,1,0,100,4,0; > /port1/port

Delivery Event

The format of the Delivery event is:

Delivery:APortName,ReasonCode[,EventData];

APortName – 64 bit hex value of the attached Port Name.

ReasonCode – hex value of the reason code.

EventData – 24 byte frame header specified as 32bit hex values separated by commas. If the event data is not to be included, the ‘;’ terminates the event after the ReasonCode.

Examples of sending FPIN with Delivery descriptor

Sending the descriptor to the fabric controller from vpi 0 of port1 with:

```
APortName – 200000110d2b4700
ReasonCode – 1 (Timeout).
EventData – 1,2,4,8,10,20
SendFpin= 0,000000,Event=Delivery:200000110d2b4700,1,1,2,4,8,10,20; > /port1/port
Sending the descriptor 4 times at 1024 second intervals:
SendFpin=0,010000,Event=LinkIntegrity:200000110d2b4700,1,0,100,4,0; > /port1/port
```

Sending the same link descriptor 3 times at 1024 second intervals (the interval of the burst and the descriptor do not need to match)

```
SendFpin=0,000000,Burst=400:4:0,Event=Delivery:200000110d2b4700,1,1,2,4,8,10,20;>
/port1/port
```

Peer Congestion Event

The format of the Peer Congestion event is:

PeerCongestion:APortName,EventType,EventModifier,EventPeriod[, PortNameList];

APortName – 64 bit hex value of the attached port name.
EventType – hex value of the event type.
EventModifier – hex value of the event modifier.
EventThreshold – hex value of the event threshold.
PortNameList – if the default portname list desired, a value of 0 is specified. If the port name list is not to be included, the ‘;’ terminates the event after the EventCount is specified. Otherwise, a comma separated list of 64-bit hex values representing the port names.

Example of sending FPIN with Peer Congestion descriptor

Sending the descriptor to the fabric controller from vpi 0 of port1 4 times at 1024 second intervals (the burst and descriptor intervals **MUST** match) with:

```
APortName – 200000110d2b4700
EventType – 2 (Credit Stall).
EventModifier – 0
EventThreshold – 400
PortNameList – 0
SendFpin=0,000000,Burst=400:4:0,Event=PeerCongestion:200000110d2b4700,2,0,400,0 >
/port1/port
```

Congestion Event

The format of the congestion event is:

Congestion:EventType,EventModifier,EventPeriod,CongestionSeverity;

EventType – hex value of the event type.
EventModifier – hex value of the event modifier.
EventThreshold – hex value of the event threshold.
CongestionSeverity – hex value of the congestion severity.

Example of sending FPIN with Peer Congestion descriptor

Sending the descriptor to the fabric controller from vpi 0 of port1 4 times at 1024 second intervals (the burst and descriptor intervals **MUST** match) with:

```
APortName – 200000110d2b4700  
EventType – 3 (Oversubscription).  
EventModifier – 0  
EventThreshold – 400  
CongestionSeverity – f7 (Alarm)  
SendFpin=0,000000,Burst=400:4:0,Event=Congestion:3,0,400,f7; > /port1/port
```

Conclusion

As shown above, SANBlaze provides a mechanism to debug and test the sending and receiving of FPINs, with various notifications in the payload that are sent to the Fabric Controller and specified N_PORTS. You can configure the sending and receiving of FPINs through both the GUI and the CLI of the software.

In addition, SANBlaze also provides logging and tracing for any FPIN sent or received. For more information, please contact us.

SANBlaze Technology, Inc.
One Monarch Drive, Suite 204
Littleton, MA 01460
(978) 679-1400
info@sanblaze.com

[Help Center](#)

[Sales](#)

[Support](#)

For General Inquiries:

[Email Sales](#) | [Email Support](#)