

Adapting the FC_Port definition to VN_Ports

Bob Nixon

bob.nixon@emulex.com

- Version 2: Added a fix for L_Port, where defined..
Removed changes to FC-LS-2, since they are covered in T11/08-379.
Split the section that affected several standards into one section per standard.
Changes marked with change bars.
- Version 1: Incorporated changes agreed at FC-FS-3 ad hoc work group meeting 2 June 2008.
Changes marked with change bars.

1 Overview

This proposal recommends a change to the definition of FC_Port to bring it into alignment with other changes recommended in pursuit of a Fibre Channel architecture that supports N_Port ID Virtualization (see T11/8-034v4 and T11/08-153v1).

2 Issue

Proposals now being considered would narrow the definitions of N_Port and Nx_Port in order to introduce the new term VN_Port. The narrowed definition of Nx_Port causes inconsistencies in standards text concerning FC_Port behavior: FC_Port is currently defined so that Nx_Ports are FC_Ports, but functions are specified for FC_Ports that are no longer within the narrower scope of Nx_Ports (e.g., sending Primitive Sequences).

As with Nx_Port, FC_Port is a term that is widely used and conceptually loaded. Any change to its definition should be carefully chosen.

3 Proposal

Redefine FC_Port to include at least one Nx_Port, but to also include other components (e.g., an LCF), so that FC_Port continues to be an object that supports FC-0, FC-1, and all sublevels of FC-2. FC_Port occupies the same architectural position as PN_Port, but occurs in any FC device, not just end devices. PN_Port becomes a type of FC_Port, and Nx_Port is a component of an FC_Port. An Nx_Port, being part of an FC_Port rather than a type of FC_Port, is not subject to FC_Port normative text other than FC-2V normative text.

This preserves the conceptual nature of FC_Port, other than its equivalence with Nx_Port.

Several changes had already been made in FC-FS-3 version 0.50 to clarify that certain behavior was particular to end devices (PN_Ports rather than FC_Ports), or independently conducted by each VN_Port within an FC_Port. A few similar changes are necessary for other standards, and are included in this proposal.

Despite that an Fx_Port/F_Port would no longer be a Fibre Channel addressable entity, no broad effort is recommended to replace these terms. This accepts that each Fx_Port/F_Port has only a single addressable entity, the F_Port Controller, and so it can be identified with the Well-known address (and name identifier) of its F_Port Controller.

4 Instructions to editor

4.1 Conventions

Deletions are indicated by ~~red-strikeout text~~. Additions are indicated by **blue text**. Comments that locate the subsequent modified text are indicated in **magenta text**.

4.2 Changes to FC-FS-3 version 0.50

In the glossary, modify the definition of FC_Port to:

3.1.29 FC_Port: A port that is capable of transmitting and receiving Fibre Channel frames according to the FC-0, FC-1, FC-2, and FC-3 levels of the Fibre Channel standards. An FC_Port includes an LCF and at least one Nx_Port. The following are FC_Ports: PN_Ports, L_Ports, F_Ports, FL_Ports, Fx_Ports, E_Ports, and B_Ports.

In the glossary, modify the definition of L_Port to:

3.1.56 L_Port: ~~A port~~**An FC_Port** that contains Arbitrated Loop functions associated with Arbitrated Loop topology (see FC-AL-2).

4.3 In FC-BB-5

3.1.x FC_Port: A port that is capable of transmitting and receiving Fibre Channel frames (see FC-FS-3).

In 3.1.13

FC_Port: A port ~~generating/terminating and/or forwarding FC frames, and generating/terminating FC Primitive Signals and Primitive Sequences. FC_Ports include N_Ports, F_Ports, E_Ports, B_Ports, VE_Ports, and B_Access~~**that is capable of transmitting and receiving Fibre Channel frames (see FC-FS-3).**

4.4 In FC-DA-2 version 1.02

In 3.2.13

FC_Port: A port ~~transmitting or receiving Fibre Channel frames. FC_Port includes any port type defined in the Fibre Channel family of standards (e.g., N_Port, F_Port, FL_Port, E_Port)~~**that is capable of transmitting and receiving Fibre Channel frames (see FC-FS-3).**

In 3.2.17

L_Port: ~~A port~~**An FC_Port** that contains Arbitrated Loop functions associated with the Arbitrated Loop topology (see FC-AL-2).

In 3.2.34

N_Port_Name: ~~An~~**A** Name_Identifier (see 3.2.19) associated with an ~~FC_Port~~**Nx_Port** (see FC-FS-2).

In table 18

- ^k A port is required to accept an RPBC if it has set the Query Data Buffer conditions bit to one in its login with the ~~FC_Port~~Nx_Port that sent the RPBC; otherwise, it may reject the RPBC.

4.5 In FC-GS-6 version 9.10

In 6.6.4.4.3.6

The Supported FC-4 Types attribute has a format identical to that of a Name Server FC-4 TYPEs object. An ~~FC_Port~~Nx_Port shall register a Supported FC-4 Types value that indicates “support” for any FC-4 TYPE that it is able to be configured to support. The Supported FC-4 Types attribute is registered with the HBA Management Server as a Port attribute.

In 6.6.4.4.3.7

The Port Active FC-4 Types attribute is registered with the Name Server as a Name Server FC-4 TYPEs object and is subject to all the description and constraints of that object. An ~~FC_Port~~Nx_Port shall register a Name Server FC-4 TYPEs value that indicates support for any FC-4 TYPE that the ~~FC_Port~~Nx_Port is completely configured to support.

In 6.6.4.4.3.17

The Number of Discovered Ports attribute returns the number of ~~FC_Ports~~Nx_Ports that are visible to the Nx_Port identified in the request. At a minimum, this is the number of ~~FC_Ports~~Nx_Ports mapped to a device, but may not reflect all of the nodes on a network.

4.6 In FC-IFR

In 3.1.12

FC_Port: A port that is capable of transmitting or receiving Fibre Channel frames (see FC-FS-3) ~~according to the requirements defined in this standard. FC_Ports include N_Ports, NL_Ports, Nx_Ports, L_Ports, F_Ports, FL_Ports, Fx_Ports, E_Ports, and B_Ports.~~

3.1.x **FC_Port:** A port that is capable of transmitting and receiving Fibre Channel frames (see FC-FS-3).

4.7 In FC-SP-2 equivalent to FC-SP rev 1.8

In the Glossary

3.2.28 Exchange: The unit of protocol activity that transfers information between a specific Originator ~~FC_Port~~Nx_Port and specific Responder ~~FC_Port~~Nx_Port using one or more related non-concurrent Sequences that may flow in the same or opposite directions. See FC-FS-2.

In 5.9.1

An AUTH_ELS transfers an Authentication message or a fragment of an Authentication message. AUTH_ELS requires Login between the two associated ~~FC_Ports~~Nx_Ports (e.g., for authentication with a Fabric, one Nx_Port is an F_Port Controller) prior to its use.

Any Nx_Port or Fx_Port/F_Port Controller may act as Authentication Initiator or as Authentication Responder. An Nx_Port or a Fx_Port/F_Port Controller may initiate an Authentication Transaction whenever appropriate (see 8). No more than one Authentication Transaction shall be in progress between a pair of Nx_Ports, or an Nx_Port and a Fx_Port/F_Port Controller, at any time.

If two FC_Ports/Nx_Ports start an Authentication Transaction at the same time, one of the two Authentication Transactions shall be aborted, as described in this subclause.

If an Nx_Port or Fx_Port/F_Port Controller is acting as an Authentication Initiator and receives an AUTH_Negotiate message from the designated Authentication Responder, one of the two Authentication Transactions shall be aborted. In the case of Nx_Port to Nx_Port Authentication, the Nx_Port that sent the AUTH_Negotiate message with the numerically higher Name shall remain the Authentication Initiator, while the Nx_Port that sent the AUTH_Negotiate message with the numerically lower Name shall become the Authentication Responder. In the case of Nx_Port to Fx_Port/F_Port Controller Authentication, the Nx_Port shall remain the Authentication Initiator, while the Fx_Port/F_Port Controller shall become the Authentication Responder. The FC_Port/Nx_Port that remains the Authentication Initiator shall reply to the received AUTH_Negotiate message with an AUTH_Reject message with Reason Code 'Logical Error' and Reason Code Explanation 'Authentication Transaction Already Started'. The FC_Port/Nx_Port that becomes the Authentication Responder shall reply to the received AUTH_Negotiate message and abort its own transaction upon receipt of the AUTH_Reject message.

In 5.9.1

Addressing: The S_ID field shall be the address identifier of the Nx_Port sending the AUTH_ELS message, or FFFFF0h (i.e., the N_Port Controller address), or FFFFFEh (i.e., the F_Port Controller address) if the AUTH_ELS message is being sent from an Fx_Port to an Nx_Port, or the Well Known Address of a Generic Service. The D_ID field shall designate the FC_Port/Nx_Port to which Authentication is being performed, and shall be one of:

- a) The address identifier of another Nx_Port to designate that Nx_Port as the FC_Port/Nx_Port to which Authentication is being performed;
- b) The N_Port Controller address (i.e., FFFFF0h) to designate a VFT capable N_Port/PN_Port (see FC-FS-2 and FC-LS);
- c) The Well Known Address of a Generic Service to designate that Generic Service as the FC_Port/Nx_Port to which Authentication is being performed; or
- d) FFFFFEh to designate the local Fx_Port/F_Port Controller as the FC_Port/Nx_Port to which Authentication is being performed. Authentication of an Nx_Port with its local Fx_Port/F_Port Controller shall be equivalent to Authentication with the Fabric.

Replace figure 26

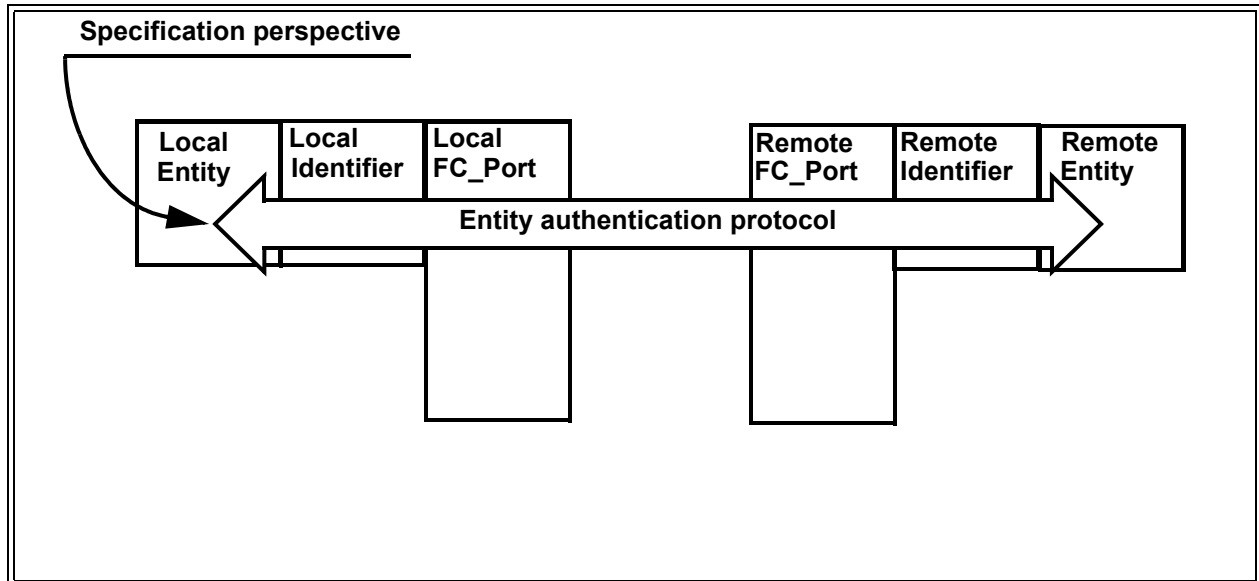


Figure 1 — Entity authentication specification perspective

with

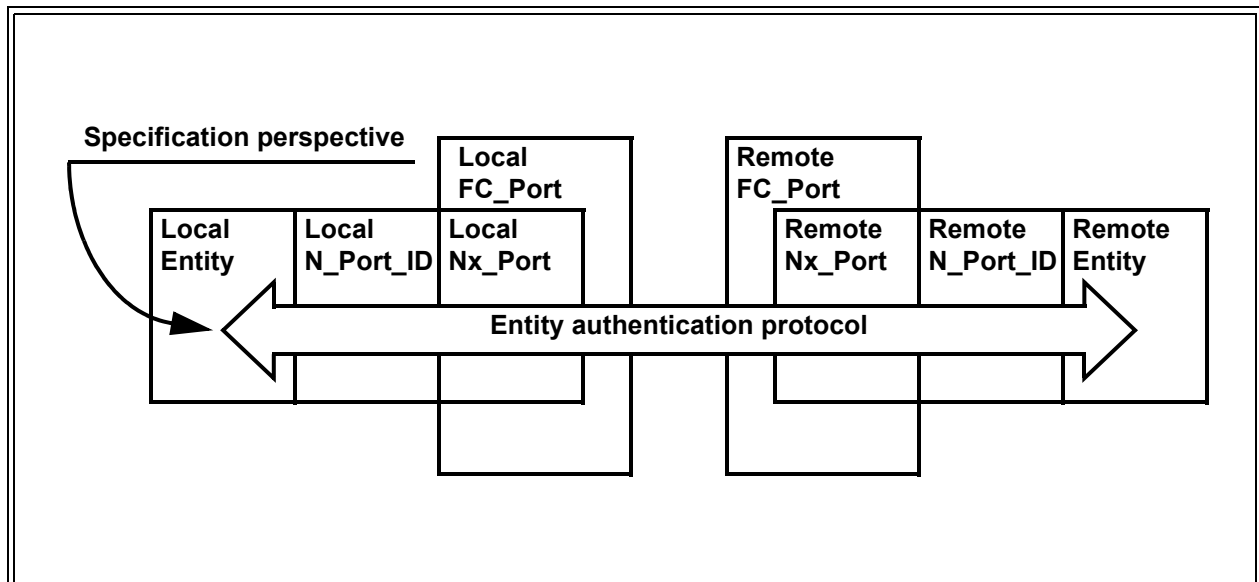


Figure 2 — Entity authentication specification perspective

In 8.5.3.3

If the Security service detects an IKEv2 dead peer condition (see 6.8.4) between a local ~~FC_Port~~Nx_Port entity and a remote ~~FC_Port~~Nx_Port entity, it shall report an IKEv2 dead peer event to the state machine.

In 8.5.4.8

The link initialization request shall cause the FC-2 service to perform link initialization for the local FC_Port in accord with FC-FS-2. This in turn shall lead to implicit fabric logout of ~~the local FC_Port if it is an Nx_Port, and implicit fabric logout of~~ all the N_Port_IDs assigned by the local FC_Port ~~if it is an Fx_Port~~.

In 8.5.4.9

The Disable request shall cause the FC-2 service to cause the local ~~FC_Port~~Nx_Port to be unable to communicate with the remote ~~FC_Port~~Nx_Port. A disabled ~~FC_Port~~Nx_Port should remain disabled pending vendor-specific outside intervention. If:

- a) either the local FC_Port or the remote FC_Port is an Fx_Port;
- b) the local FC_Port is not operating as an L_Port;
- c) the local FC_Port does not support additional N_Port_ID assignment; and
- d) the local FC_Port does not support Virtual Fabric Tagging,

the local ~~FC_Port~~Nx_Port may be disabled by holding a receiver reset condition on the receiver (see FC-FS-2) and holding the transmitter in the not-enabled state (see FC-PI-2), or by refusing to initiate or respond to FLOGI (see FC-FS-2). Otherwise, the local ~~FC_Port~~Nx_Port may be disabled by refusing to either initiate or respond to FLOGI or FDISC or PLOGI with the remote ~~FC_Port~~Nx_Port, but shall not be disabled by holding a receiver reset condition on the receiver or holding the transmitter in the not-enabled state.

In 8.9.4

FC-LS defines a state in which to perform N_Port to F_Port Authentication before the EVFP protocol (see FC-LS). This state is called 'P2:AUTH_ELS'. In this state an AUTH_ELS transaction (see 5.9) between the involved N_Port and F_Port shall be performed. If the two involved FC_Ports negotiated to perform the EVFP protocol (see FC-LS) then the identities authenticated shall be the Core Switch_Name for the F_Port (see FC-SW-4) and the Core N_Port_Name or the Node_Name for the N_Port (see FC-FS-2). If the two involved FC_Ports negotiated to not perform the EVFP protocol (see FC-LS) then the identities authenticated shall be the Core Switch_Name or the Switch_Name associated with the Port VF_ID for the F_Port (see FC-SW-4) and the Core N_Port_Name or the Node_Name or the N_Port_Name of the ~~Logical N_Port~~VN_Port associated with the Port VF_ID for the N_Port (see FC-FS-2).