Brian Hart/Austin/IBM
03/07/2008 01:59 PM

Default custom expiration date of 03/07/2009

To Kevin D Butt/Tucson/IBM@IBMUS
cc Jerry Poole/Raleigh/IBM@IBMUS, Christine R Knibloe/Tucson/IBM@IBMUS, Khuong Pham/Austin/IBM@IBMUS, Dan
bcc
Subject FCP-4 defect reports

Kevin,

I would like to report the following issues with FCP-4.

All references are to FCP-4 r00a, except where noted. All references to FC-FS-2 are to FC-FS-2 draft revision 1.01. Subscripts are rendered by concatenation and lower-case (e.g. using "R_A_TOVels" in place of "R_A_TOV" with a subscript "ELS").

The arguments dealing with timers and recovery all assume a fabric environment and unacknowledged class of service.

1) Target requirement for FCP_RESID_UNDER is missing

Problem:

There is no requirement for a target to set FCP_RESID_UNDER if a read operation results in the transfer of fewer than FCP_DL bytes. The 4th paragraph of section 9.4.2, requires: "Because there were fewer bytes provided than required by FCP_DL, the FCP_RESID_UNDER bit...shall be set to one in the FCP_RSP IU...." But this occurs in the context of a discussion of a write operation. There is no similar requirement that FCP_RESID_UNDER be set appropriately in the context of read operations.

Section 12.2.2 first paragraph bullet (b) requires the initiator to detect underrun. This may imply a requirement for the target, but it would be better explicitly stated.

Proposed resolution:

- Break section 9.4 paragraph 4 after "...the target FCP_Port shall discard the excess bytes.", -and-

- Amend the following sentence to replace "Because there were fewer

bytes provided than required...." with "If an operation results in the
transfer of fewer bytes than required....".

2) Timer summary table is unclear

Problem:

The timer summary table (Table 30) contains a column headed "Default
Value". In some cases the column contains a description of a range,
rather than a value; the column header is misnamed.

The use of ranges in this column suggests that constraints are being
expressed, but this is not stated in the text. "Default value" is not
defined by the standard so is left to assume its normal English
meaning. "Default" then suggests that implementations are free to
choose a different value--including one not in the suggested range. It
is ambiguous whether the table states a constraint.

Proposed resolution:

- Replace all range definitions in the "Default Value" column with
values. All the current ranges are specified as ">="; use the floor
of the range as the specified default value. -and-

- Add a column "Range" to the table; express the allowable ranges in
this column. -and-

- Add the following text to section 11.1: "FCP_Ports should use the
default values specified in table 30 for those timers. If an FCP_Port
chooses or negotiates a different value for a timer, the value shall
fall in the range specified in the table."

3) "Sequence level recovery" is not defined

Problem:

Every usage of the phrase "Sequence level recovery" has the indicated
capitalization. This is a marked usage and suggests that the phrase is
being used as a term of art. However, the phrase is not defined by the

standard, so is left to assume its normal English meaning.

It is not clear how the normal meaning of the phrase relates to the concepts of the standard. Specifically, it is not clear when an FCP_Port "ha[s] agreed to Sequence level recovery". What constitutes this agreement should be clearly defined as it qualifies several sections describing recovery. This has ramifications for data integrity (see, e.g., issue (4) below).

Proposed resolution:

- In section 6.3.4, subsection "Word 3, Bit 8: RETRY", add a sentence following the first sentence of the third paragraph: "....in both the request payload and in the accept payload. In this case the initiator and target shall have agreed to Sequence level recovery."

4) Recovery is insufficiently required

Problem:

Several recovery sections (e.g. 12.4.1.5) are qualified by: "This procedure shall be used only by FCP devices that have agreed to Sequence level recovery". That is, agreement to Sequence level recovery is necessary but not sufficient to imply that an initiator or target will perform the defined recovery. The standard provides no mechanism for an agreeable FCP_Port to communicate its actual intent to follow the recovery procedures, so it is possible that an initiator and target might make opposite choices.

There are cases, though, where either both or neither initiator and target must perform the recovery in order to preserve data integrity.

A target, for example, might agree to Sequence level recovery but elect not to perform the FCP_RSP IU recovery described in section 12.4.1.5. Not being subject, then, to the restrictions in 12.4.1.5, the target would be at liberty to discard exchange information as soon as an

FCP_RSP was sent. If the FCP_RSP were lost, an otherwise timely REC
by the initiator would be rejected by the target with "Logical error"/
Invalid OX_ID-RX_ID combination". The initiator could then resend the
FCP_CMND (per 12.4.1.3) to the detriment of data integrity. (The
target would have performed the operation twice but the initiator would
believe that it had only been performed once.)

Proposed resolution:

- Replace the qualifications at the heads of sections 12.4.1.3,
12.4.1.4, 12.4.1.5, 12.4.1.6, and 12.4.1.7 with: "This procedure shall
be used by and only by FCP devices that have agreed to Sequence level
recovery." Note the larger effect on 12.4.1.3 than on the others.

5) R_A_TOV (re)definitions drop vital guarantee

Problem:

Section 11.3 states: "R_A_TOV has two separate components, labeled
R_A_TOVseq_qual and R_A_TOVels." FC-FS-2 contains no mention of
separate components of R_A_TOV. It's unclear whether FCP's R_A_TOV
component timers inherit substance or merely name from FC-FS-2.

FC-FS-2 section 20.2.1.4 provides a guarantee: "R_A_TOV represents
E_D_TOV plus twice the maximum time that a frame may be delayed within
a Fabric and still be delivered." The notion that R_A_TOV encompasses
the maximum fabric delivery time is vital to the definition of
RR_TOVseq_init (Table 30) and the recovery mechanisms that depend on it
(e.g. section 12.4.1.5).

If R_A_TOVels does not inherit substantially from FC-FS-2 R_A_TOV then
this vital guarantee is dropped. Even if R_A_TOVels does inherit
substantially from FC-FS-2 R_A_TOV, Table 30 flatly redefines the
duration of R_A_TOVels as 2 or 10 seconds without mention of maximum
fabric delivery time, dropping the vital guarantee.

Proposed resolution:

- Amend Table 30 - Timer summary NOTE 1 to add a sentence: "R_A_TOV
for ELS shall encompass the maximum time that a frame may be delayed
within a Fabric and still be delivered."

Note that boundedness of R_A_TOVels directly affects boundedness of
RR_TOVseq_init, and so has implications for boundedness of REC_TOV.
See (7) below.

6) REC_TOV floor allows REC vs FCP_CMND race

Problem:

Section 12.4.1.3 equates REC reject (with "Logical error"/"Invalid
OX_ID-RX_ID combination") to the loss of the FCP_CMND and prescribes
retransmission of the FCP_CMND. But an initiator would see the same
reject in the case where the REC merely arrived at the target ahead of
the FCP_CMND. In that case retransmission of the FCP_CMND could result
in a loss of data integrity.

Arrival of REC ahead of FCP_CMND could be prevented by ensuring that
REC is not transmitted until it is certain that the FCP_CMND is either
delivered or lost.

FC-FS-2 section 20.2.1.3 limits to three the actions whose duration is
bounded by E_D_TOV; frame delivery across a fabric is not among those.
Rather, FC-FS-2 section 20.2.1.4 describes R_A_TOV as the timer that
encompasses the maximum frame delivery time.

In order to ensure REC is not sent prematurely, REC_TOV's range must
therefore encompass R_A_TOV rather than E_D_TOV.

Proposed resolutions:

- Replace REC_TOV range of ">= E_D_TOV + 1s" with ">= R_A_TOV" in Table
30 - Timer summary. -or-

- Replace section 12.4.1.3 paragaph 2 with: 'If the target reports the

reports the
exchange invalid (i.e. the initiator FCP_Port receives an LS_RJT for
the REC with the reason code of "Logical error" and reason code
explanation set to "Invalid OX_ID-RX_ID combination"), the initiator
shall not retransmit the FCP_CMND and shall notify the application
client appropriately.'

Note that if 12.4.1.3 is allowed to stand a modification may still be
in order. Verb "retransmit" following the parenthetical is in the
imperative mood and would better be declarative: "...), the initiator
shall retransmit...."

7) Lack of REC_TOV ceiling allows REC vs exchange discard race

Problem:

REC_TOV is described in the timer summary table (Table 30) as a range
with a floor but no ceiling. No mechanism is provided to communicate
the choice of REC_TOV between initiator and target. This allows the
possibility that an initiator may choose a REC_TOV that is arbitrarily
large and that differs from the REC_TOV chosen by the target.

Further, section 11.5 describes REC_TOV as the "minimum polling
interval" for REC and states that a duration of "at least" REC_TOV
occurs before REC may be sent. REC_TOV is not a ceiling on the REC
polling interval.

Section 12.4.1.5 attempts to ensure that a target will maintain
exchange information until a timely REC arrives by requiring that the
target retain the information for up to RR_TOVseq_init after sending
the FCP_RSP.

Table 30 suggests RR_TOVseq_init should be ">= REC_TOV + 2xR_A_TOVels
+ 1s" (in the RETRY case), but this is insufficient. The target must
necessarily base its RR_TOVseq_init on its own REC_TOV since it has no

knowledge of the initiator's REC_TOV. The initiator's REC_TOV can be
arbitrarily larger than the target's, so the target can be left with an
RR_TOVseq_init that does not encompass the initiator's REC_TOV.

Even when the initiator and target have sufficiently similar REC_TOV,
the initiator may delay arbitrarily beyond REC_TOV before transmiting
the REC, leaving the target with an RR_TOVseq_init that does not
encompass the initiator's REC polling interval.

If the initiator sends REC after the target's RR_TOVseq_init expires
(or merely late enough in the RR_TOVseq_init interval), the REC will
(may) arrive after RR_TOVseq_init has expired. The target, then, may
have discarded the exchange information in accordance with 12.4.1.5 and
will reject the REC with "Logical error"/"Invalid OX_ID-RX_ID
combination". The initiator may respond by resending the FCP_CMND
(per 12.4.1.3) to the detriment of data integrity.

The initiator's REC polling interval must be constrained to ensure the
REC arrives at the target before the expiration of RR_TOVseq_init.
This requires a ceiling on REC polling (and so also on REC_TOV) and an
effective floor on RR_TOVseq_init.

Proposed resolution:

All three of:

- Modify section 11.5 first paragraph to add a sentence encouraging
prompt polling by initiators: "....first polling for Exchange status
with the REC ELS. Initiators should transmit REC promptly after
REC_TOV expiration. Table 31...." -and-

- Modify Table 30 to set an appropriate ceiling for REC_TOV, perhaps
one of: "<= R_A_TOV", "<= R_A_TOV + E_D_TOV", or "<= 2xR_A_TOV". -and-

- Modify Table 30 to set a floor for RR_TOVseq_init based on the
REC_TOV ceiling, making RR_TOVseq_init's range: ">= ceil(REC_TOV) +

R_A_TOV + 1s" (with "R_A_TOV" allowing time for the REC to traverse the
fabric and "1s" as an allowance for initiator promptness failings).

Or just:

- Replace section 12.4.1.3 paragaph 2 with: 'If the target reports the
exchange invalid (i.e. the initiator FCP_Port receives an LS_RJT for
the REC with the reason code of "Logical error" and reason code
explanation set to "Invalid OX_ID-RX_ID combination"), the initiator
shall not retransmit the FCP_CMND and shall notify the application
client of the problem.'

Note that if 12.4.1.3 is allowed to stand a modification may still be
in order. Verb "retransmit" following the parenthetical is in the
imperative mood and would better be declarative: "...), the initiator
shall retransmit...."

Regards,
brian
--
Brian Hart IBM AIX Support
SAN Team
hartb@us.ibm.com
512-823-7856