1. Revision history
   Revision 0 (11 September 2007) First revision

2. Related documents
   spc4r11 – SCSI Primary Commands - 4

3. Overview
   The text in the “Basic PERSISTENT RESERVE OUT parameter list” section
   (6.14.3) of the PERSISTENT RESERVE OUT command which describes how to
   use the PR-OUT command to unregister a key, or to change the registration key is
   inconsistent with the text in the table 34 (in section 5.6.6 in the model clause).
   One section of the text incorrectly indicates that the RESERVATION KEY field
   is used by the REGISTER AND IGNORE EXISTING KEY service action, while
   a second section of the text correctly states that the RESERVATION KEY field is
   ignored by the REGISTER AND IGNORE EXISTING KEY service action. The
   tables (33 and 34) are also correct.
   This proposal corrects the incorrect section of the text to make it consistent with
   the other correct sections.

   <Red Text> indicates changes.

Proposal:
### Table 33 — Register behaviors for a REGISTER service action

<table>
<thead>
<tr>
<th>Command I_T nexus status</th>
<th>Parameter list fields</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>received on an unregistered I_T nexus</td>
<td></td>
<td>Do nothing except return GOOD status.</td>
</tr>
<tr>
<td></td>
<td>RESERVATION KEY: zero</td>
<td>SERVICE ACTION RESERVATION KEY: zero</td>
</tr>
<tr>
<td></td>
<td>RESERVATION KEY: non-zero</td>
<td>SERVICE ACTION RESERVATION KEY: one</td>
</tr>
<tr>
<td></td>
<td>RESERVATION KEY: non-zero</td>
<td>SERVICE ACTION RESERVATION KEY: zero</td>
</tr>
<tr>
<td>received on a registered I_T nexus</td>
<td></td>
<td>Unregister the I_T nexus on which the command was received (see 5.6.10.3).</td>
</tr>
<tr>
<td></td>
<td>RESERVATION KEY: not equal to I_T nexus reservation key</td>
<td>SERVICE ACTION RESERVATION KEY: zero</td>
</tr>
<tr>
<td></td>
<td>RESERVATION KEY: equal to I_T nexus reservation key</td>
<td>SERVICE ACTION RESERVATION KEY: one</td>
</tr>
<tr>
<td></td>
<td>RESERVATION KEY: non-zero</td>
<td>SERVICE ACTION RESERVATION KEY: zero</td>
</tr>
<tr>
<td></td>
<td>RESERVATION KEY: non-zero</td>
<td>SERVICE ACTION RESERVATION KEY: one</td>
</tr>
</tbody>
</table>

* For requirements regarding the parameter list fields not shown in this table see 6.14.3.
* If any I_T nexus specified in the parameter list is registered, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN PARAMETER LIST. Devices compliant with previous versions of this standard may return an additional sense code set to INVALID FIELD IN CDB.
* The sense key shall be set to ILLEGAL REQUEST, and the additional sense code shall be set to INVALID FIELD IN PARAMETER LIST. Devices compliant with previous versions of this standard may return an additional sense code set to INVALID FIELD IN CDB.
6.14.3 Basic PERSISTENT RESERVE OUT parameter list

The RESERVATION KEY field contains an 8-byte value provided by the application client to the device server to identify the I_T nexus that is the source of the PERSISTENT RESERVE OUT command. The device server shall verify that the contents of the RESERVATION KEY field in a PERSISTENT RESERVE OUT command parameter data matches the registered reservation key for the I_T nexus from which the command was received, except for:

a) The REGISTER AND IGNORE EXISTING KEY service action where the RESERVATION KEY field shall be ignored; and
b) The REGISTER service action for an unregistered I_T nexus where the RESERVATION KEY field shall contain zero.

Except as noted above, when a PERSISTENT RESERVE OUT command specifies a RESERVATION KEY field other than the reservation key registered for the I_T nexus the device server shall return a RESERVATION CONFLICT status. Except as noted above, the reservation key of the I_T nexus shall be verified to be correct regardless of the SERVICE ACTION and SCOPE field values.

The SERVICE ACTION RESERVATION KEY field contains information needed for the following service actions: REGISTER, REGISTER AND IGNORE EXISTING KEY, PREEMPT, and PREEMPT AND ABORT. The SERVICE ACTION RESERVATION KEY field is ignored for the following service actions: RESERVE, RELEASE, and CLEAR.

For the REGISTER service action and REGISTER AND IGNORE EXISTING KEY service action, the SERVICE ACTION RESERVATION KEY field contains:
a) The new reservation key to be registered in place of the registered reservation key specified in the RESER-
VATION KEY field; or
b) Zero to unregister the registered reservation key specified in the RESERVATION KEY field.

For the REGISTER AND IGNORE EXISTING KEY service action, the SERVICE ACTION RESERVATION KEY field
contains:

a) The new reservation key to be registered in place of the registered reservation key; or
b) Zero to unregister the registered reservation key.

For the PREEMPT service action and PREEMPT AND ABORT service action, the SERVICE ACTION RESERVATION
KEY field contains the reservation key of:

a) The registrations to be removed; and
b) If the SERVICE ACTION RESERVATION KEY field identifies a persistent reservation holder (see 5.6.9), persistent
reservations that are to be preempted.