UNH–IOL
NVMe Testing Service

Test Plan for NVMe
Subsystem Local Memory Command Set
Conformance
Version 22.0
Target Specification:
NVMe Subsystem Local Memory
Command Set Specification 1.0
Technical Document

Last Updated: July31, 2024

UNH–IOL NVMe Testing Service
21 Madbury Rd Suite 100
Durham, NH 03824

Tel: +1 603–862–0090
Fax: +1 603–862–4181
Email: nvmelab@iol.unh.edu

TABLE OF CONTENTS

MODIFICATION RECORD

2024 July 8 (Version 22.0) Initial Release

Carter Snay:

1. Initial release of the NVMe Subsystem Local Memory Command Set Conformance Test Suite with v22 of the NVMe ICC test plans.

ACKNOWLEDGMENTS

The UNH–IOL would like to acknowledge the efforts of the following individuals in the development of this test plan:

Carter Snay	UNH InterOperability Laboratory
Michael Crawley	UNH InterOperability Laboratory
Liam Warren	UNH InterOperability Laboratory

INTRODUCTION

The University of New Hampshire's InterOperability Laboratory (IOL) is an institution designed to improve the interoperability of standards–based products by providing a neutral environment where a product can be tested against other implementations of a common standard, both in terms of interoperability and conformance. This particular suite of tests has been developed to help implementers evaluate the NVMe functionality of their products. This test suite is aimed at validating products in support of the work being directed by the NVMe Promoters Group.

These tests are designed to determine if a product conforms to specifications defined in the NVM Express Subsystem Local Memory Specification Revision 1.0 specification and NVM Express Base Specification 2.0d. Successful completion of these tests provides a reasonable level of confidence that the Device Under Test (DUT) will function properly in many NVMe environments.

The tests contained in this document are organized to simplify the identification of information related to a test, and to facilitate the actual testing process. Tests are separated into groups, primarily to reduce setup time in the lab environment, however the different groups typically also tend to focus on specific aspects of device functionality. A two–number, dot–notated naming system is used to catalog the tests. This format allows for the addition of future tests in the appropriate groups without requiring the renumbering of the subsequent tests.

The test definitions themselves are intended to provide a high–level description of the motivation, resources, procedures, and methodologies specific to each test. Formally, each test description contains the following sections:

Purpose
The purpose is a brief statement outlining what the test attempts to achieve. The test is written at the functional level.

References
This section specifies all reference material *external* to the test suite, including the specific references for the test in question, and any other references that might be helpful in understanding the test methodology and/or test results. External sources are always referenced by a bracketed number (e.g., [1]) when mentioned in the test description. Any other references in the test description that are not indicated in this manner refer to elements within the test suite document itself (e.g., "Appendix 5.A", or "Table 5.1.1–1").

Resource Requirements
The requirements section specifies the test hardware and/or software needed to perform the test. This is generally expressed in terms of minimum requirements, however in some cases specific equipment manufacturer/model information may be provided.

Last Modification
This specifies the date of the last modification to this test.

Discussion
The discussion covers the assumptions made in the design or implementation of the test, as well as known limitations. Other items specific to the test are covered here as well.

Test Setup
The setup section describes the initial configuration of the test environment. Small changes in the configuration should not be included here and are generally covered in the test procedure section (next).

Procedure

The procedure section of the test description contains the systematic instructions for carrying out the test. It provides a cookbook approach to testing, and may be interspersed with observable results. These procedures should be the ideal test methodology, independent of specific tool limitations or restrictions.

Observable Results
This section lists the specific observable items that can be examined by the tester in order to verify that the DUT is operating properly. When multiple values for an observable are possible, this section provides a short discussion on how to interpret them. The determination of a pass or fail outcome for a particular test is generally based on the successful (or unsuccessful) detection of a specific observable.

Possible Problems
This section contains a description of known issues with the test procedure, which may affect test results in certain situations. It may also refer the reader to test suite appendices and/or other external sources that may provide more detail regarding these issues.

REFERENCES

The following documents are referenced in this text:

1. NVM Express Subsystem Local Memory Command Set Specification Revision 1.0 (December 20, 2023)
2. NVM Express Base Specification Revision 2.0d (January 11, 2024)

ABBREVIATIONS

The following abbreviations are applied to the test titles of each of the tests described in this document for indicating the status of test requirements.

M - Mandatory
FYI - FYI
IP - In Progress

The following abbreviations applied to the test titles of each of the tests described in this document for indicating what product types a test may apply to. It is assumed that all tests apply to base NVMe products using PCIe.

Group 1: Subsystem Local Memory Admin Command Sett

Overview:

This section describes a method for performing conformance verification for NVMe products implementing the Admin Command Set..

Notes:

The preliminary draft descriptions for the tests defined in this group are considered complete, and the tests are pending implementation (during which time additional revisions/modifications are likely to occur).

Test 1.1 – Identify Command (FYI)

Purpose: To verify that an NVMe Controller can properly execute an Identify command.

References:

[1] NVM Express Subsystem Local Memory Command Set Specification 1.0 : 4.1.6

Resource Requirements:

Tools capable of monitoring and decoding traffic on the NVMe interface.

Last Modification: July 9, 2024

Discussion: The Identify command returns a data buffer that describes the NVM subsystem, the controller or the namespace(s). The data structure is 4096 bytes in size. The host indicates as a command parameter whether to return the controller or namespace specific data structure. For the namespace data structure, the definition of the structure is specific to the I/O command set selected for use.

The data structure returned is based on the Controller or Namespace Structure (CNS) field. If there are fewer namespace identifiers or controller identifiers to return for a Namespace List or Controller List, respectively, then the unused portion of the list is zero filled. Controllers that support Namespace Management shall support CNS values of 10h–13h.

The Identify command uses the PRP Entry 1, PRP Entry 2, and Command Dword 10 fields. All other Command specific fields are reserved. A completion queue entry is posted to the Admin Completion Queue when the Identify data structure has been posted to the memory buffer indicated in PRP Entry 1.

Test Setup: See Appendix A.

Case 1: I/O Command Set Specific Identify Namespace (CNS 05h, CSI 03h) (FYI)

Test Procedure:

Test Procedure:

1. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 03h).

Observable Results:

1. Verify that the entry for NSZE in the data structure returned in step 1 has its least 2 significant bits cleared to 00h.
2. Verify that the entry for the NF field is less than 32.
3. Verify that the Format Data Structure for SLM Format 0 Support reports a value of 1 in its VAL field.
4. Verify that for all SLM Formats the DS field is cleared to 0..
5. Verify that all SLM Formats Supports other than SLM Format 0 Support report a VAL field cleared to 0 in their Format Data Structure.

Case 2: I/O Command Set Specific Identify Namespace, MCMCL and MCMSSRL ≠ 0 (CNS 05h, CSI 03h) (FYI)

Test Procedure:

1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure (LID 05h) for CSI=03h. If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.
2. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 03h).

Observable Results:

1. Verify that the Get Log command in step 1 completed successfully.
2. Verify that the values for MCMCL and MCMSSRL in the data structure returned by the Identify command in step 2 report non-zero values.

Case 3: I/O Command Set Specific Identify Controller (CSI 03h) (FYI)

Test Procedure:

1. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set Specific Identify Controller data structure (CNS 06h, CSI 03h).

Observable Results:

1. Verify that the VER field of the data structure returned by the Identify Command in step 1 reports a value of 1h in the MJR field, and 0h in the MNR and TER fields.

Test 1.2 – SLM Get Log Page Command (FYI)

Purpose: To verify that an NVMe Controller can properly execute an Identify command for SLM specific log pages.

References:

[1] NVM Express Subsystem Local Memory Command Set Specification 1.0 : 4.1.5

Resource Requirements:

Tools capable of monitoring and decoding traffic on the NVMe interface.

Last Modification: July 8, 2024

Discussion: The Get Log Page command returns a data buffer that contains the log page requested. The Get Log Page command uses the PRP Entry 1, PRP Entry 2, and Command Dword 10 fields. All other command specific fields are reserved.

The NVM Express Subsystem Local Memory Command Set Specification does not define any new log pages that are not included in the NVM Express Base Specification. There are two mandatory log pages that a device supporting the NVM Express Subsystem Local Memory Command Set are required to support. They are shown in table 1 below:

Table 1 – Log Page Identifiers

Log Identifier	Optional/ Mandatory	Log Page
1Ah	Mandatory	Reachability Groups
1Bh	Mandatory	Reachability Associations

Test Setup: See Appendix A.

Case 1: Reachability Support (FYI)

Test Procedure:

1. Configure the NVMe Host to issue a Get Log command with an LID of 00h for the Supported Log Pages.
2. Configure the NVMe Host to issue an Identify command with a CNS of 01h for the Identity Controller data structure.
3. Configure the NVMe Host to issue a Get Log command with an LID of 1Ah for the Reachability Groups log page.
4. Configure the NVMe Host to issue a Get Log command with an LID of 1Bh for the Reachability Associations log page.

Observable Results:

1. Verify all commands completed successfully.
2. Verify that the entries in the Supported Logs Page for LIDs 1Ah and 1Bh report a value of 1b in their LSUPP fields.
3. Verify that bit 0 of CRCAP (RRSUP) from the Identify Controller data structure is set to 1b.

Group 2: Subsystem Local Memory Command Set Specific I/O Commands

Test 2.1 – Memory Copy Command (FYI)

Purpose: To verify that an NVMe Controller is able to successfully copy data between source ranges within or across namespaces.

References:

[1] NVM Express Subsystem Local Memory Command Set Specification 1.0 : 3.2.1

Resource Requirements:

Tools capable of monitoring and decoding traffic on the NVMe interface.

Last Modification: July 8, 2024

Test Setup: See Appendix A.

Discussion: The Memory Copy commands used by the host to copy data from one or more source ranges in one or more source namespaces to a single consecutive address range in a destination memory namespace. The tests in this group cycle through the existing Copy command formats, both when they are enabled and disabled. The cases in this group also attempt to send the commands with different valid and invalid command parameters to ensure the controller is able to properly handle error conditions, and also perform as expected when the parameters are expected to be valid.

Case 1: Memory Copy Command, Format 2h (FYI)

Test Procedure:

1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure for SLM commands (LID 05h, CSI 03h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.
2. Configure the NVMe Host to issue an Identify command to Identify Controller (CNS 1h). If bit 2 of the Copy Descriptor Formats Supported field indicates a lack of support for Descriptor Format 2h, this test is not applicable.
3. Determine if the DUT has an NVM namespace attached to the controller. If the DUT does not have an NVM namespace, issue a Namespace Management command to create one and then a Namespace Attachment command to attach it to the controller.
4. Determine the Reachability of the NVM namespace and the memory namespace. If there is no Reachability Association between the two namespaces this test is not applicable.
5. Configure the NVMe Host to issue an Identify command to obtain the Identify Namespace data structure (CNS 00h) for the NVM namespace. Determine the block size of the device for the selected format 2^LBADS).
6. Configure the NVMe Host to issue a Set Feature command to FID 16h to set Copy Descriptor Format 2h Enable to 1h.
7. Configure the NVMe Host to issue a Write command to write 2^LBADS*2 bytes (2 blocks) of FFs to SLBA 0 on the NVM namespace.
8. Configure the NVMe Host to issue a Memory Write command to write 2^LBADS*2 bytes of 00s to SB 0 of the memory namespace.
9. Configure the NVMe Host to issue a Memory Copy command to the memory namespace with LEN=2^LBADS*2, SDADDR=0, Descriptor Format=2h, NR=0h, and a Source Range Entry containing the following values:
    a. SNSID=NSID of NVM namespace;
    b. Starting LBA=0h;
    c. NLB=2h; and

d. FCO=0h

10. Configure the NVMe Host to issue a Memory Read command to read the byte range written to in step 8.

Observable Results:

1. Verify all commands completed successfully
2. Verify that the Copy Command Format Descriptor Format 2h was enabled.
3. Verify that the Read command from step 10 returns 2^LBADS*2 bytes of FFs which were copied from the NVM namespaces to the memory namespace in step 9.

Case 2: Memory Copy Command, Format 2h (FYI)

Test Procedure:

1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure for SLM commands (LID 05h, CSI 03h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.
2. Configure the NVMe Host to issue an Identify command to obtain the Identify Controller data structure (CNS 1h). If bit 3 of the Copy Descriptor Formats Supported field indicates a lack of support for Descriptor Format 3h, this test is not applicable.
3. Determine if the DUT has an NVM namespace attached to the controller. If the DUT does not have an NVM namespace, issue a Namespace Management command to create one and then a Namespace Attachment command to attach it to the controller.
4. Determine the Reachability of the NVM namespace and the memory namespace. If there is no Reachability Association between the two namespaces this test is not applicable.
5. Configure the NVMe Host to issue an Identify command to obtain the Identify Namespace data structure (CNS 00h) for the NVM namespace. Determine the block size of the device for the selected format (2^LBADS).
6. Configure the NVMe Host to issue a Set Feature command to FID 16h to set Copy Descriptor Format 3h Enable to 1h.
7. Configure the NVMe Host to issue a Write command to write 2^LBADS*2 bytes (2 blocks) of FFs to SLBA 0 on the NVM namespace.
8. Configure the NVMe Host to issue a Memory Write command to write 2^LBADS*2 bytes of 00s to SB 0 of the memory namespace.
9. Configure the NVMe Host to issue a Memory Copy command to the memory namespace with LEN=2^LBADS*2, SDADDR=0, Descriptor Format=3h, NR=0h, and a Source Range Entry containing the following values:
    a. SNSID=NSID of NVM namespace;
    b. Starting LBA=0h;
    c. NLB=1h; and
    d. FCO=0h
10. Configure the NVMe Host to issue a Memory Read command to read the byte range written to in step 8.

Observable Results:

1. Verify all commands completed successfully.
2. Verify that the Copy Command Format Descriptor Format 3h was enabled.
3. Verify that the Read command from step 10 returns 2^LBADS*2 bytes of FFs which were copied from the NVM namespaces to the memory namespace in step 9.

Case 3: Memory Copy Command, Format 4h (FYI)

Test Procedure:

1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure for SLM commands (LID 05h, CSI 03h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.
2. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 03h). Record the NOWG field.
3. Configure the NVMe Host to issue a Set Feature command to FID 16h to set Copy Descriptor Format 4h Enable to 1h.
4. Configure the NVMe Host to issue a Memory Write command to write 2^NOWG*2 bytes of FFs to SB 0.
5. Configure the NVMe Host to issue a Memory Write command to write 2^NOWG*2 bytes of AAs to SB 2^NOWG*2.
6. Configure the NVMe Host to issue a Memory Copy command with LEN=2^NOWG*2, SDADDR=2^NOWG*2, Descriptor Format=4h, NR=0h, and a Source Range Entry containing the following values:
    a. SNSID=NSID of the memory namespace;
    b. SADDAR=0h;
    c. NBYTE=2^NOWG*2; and
    d. FCO=0h.
7. Configure the NVMe Host to issue a Memory Read command to read 2^NOWG*4 bytes starting at SB 0.

Observable Results:
1. Verify all commands completed successfully.
2. Verify that the Copy Command Format Descriptor Format 4h was enabled.
3. Verify that the Memory Read command in step 7 returned 2^NOWG*4 bytes of FFs.

Case 4: Memory Copy Command, Format Descriptors Disabled (FYI)

Test Procedure:
1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure for SLM commands (LID 05h, CSI 03h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.
2. Configure the NVMe Host to issue an Identify command to obtain the Identify Controller data structure (CNS 1h). Record all supported Copy Descriptor Formats.
3. Configure the NVMe Host to issue a Set Feature command to FID 16h to disable all supported Copy Descriptor Formats.
4. For each disabled Copy Descriptor Format, configure the NVMe Host to issue a Memory Copy command specifying the disabled format.

Observable Results:
1. Verify the Copy Descriptors Formats for each supported Format were disabled.
2. Verify all Memory Copy commands completed with the status ‘Invalid Field’.
3. Verify all other commands completed successfully.

Case 5: Memory Copy Command, Format Descriptors Not Supported (FYI)

Test Procedure:
1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure for SLM commands (LID 05h, CSI 03h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.
2. Configure the NVMe Host to issue a Memory Copy command specifying Descriptor Format Fh.

Observable Results:

1. Verify the Memory Copy command completed with the status 'Invalid Field in Command'.
2. Verify all other commands completed successfully.

Case 6: Memory Copy Command, SDADDR & LEN, Not Dword Granular (FYI)

Test Procedure:

1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure for SLM commands (LID 05h, CSI 03h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.
2. Configure the NVMe Host to issue a Set Feature command to FID 16h to set Copy Descriptor Format 4h Enable to 1h.
3. Configure the NVMe Host to issue a Memory Copy command with LEN=4h, SDADDR=5h, Descriptor Format=4h, NR=0h, and a Source Range Entry with SADDAR=0h and NBYTE=4h.
4. Configure the NVMe Host to issue a Memory Copy command with LEN=1h, SDADDR=4h, Descriptor Format=4h, NR=0h, and a Source Range Entry with SADDR=0h and NBYTE=1h.

Observable Results:

1. Verify both Memory Copy commands completed with the status 'Invalid Field'.
2. Verify all other commands completed successfully

Case 7: Memory Copy Command, NVM Namespace Not Reachable (FYI)

Test Procedure:

1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure for SLM commands (LID 05h, CSI 3h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.
2. Check the Reachability Associations and Reachability Groups log pages to determine if there is an NVM namespace which does not share a Reachability Association to a memory namespace, if none exist, then this test is not applicable.
3. Configure the NVMe Host to issue an Identify command to obtain the Identify Controller data structure (CNS 1h). If either Copy Descriptor Formats 2h or 3h are supported, select one to use for this test. If neither Copy Descriptor Formats are supported this test is not applicable.
4. Configure the NVMe Host to issue a Set Feature command to FID 16h to enable the Copy Descriptor Format determined in step 3.
5. Configure the NVMe Host to issue a Memory Copy command with the descriptor format enabled in step 4, NSID set to the memory namespace and the SNSID in the Source Range Entry set to the NVM namespace which is not reachable to the first.

Observable Results:

1. Verify the Memory Copy command in step 5 completed with the status 'Namespace Not Reachable'.
2. Verify all other commands completed successfully.

Case 8: Memory Copy Command, Memory Namespace Not Reachable (FYI)

Test Procedure:

1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure for SLM commands (LID 05h, CSI 3h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.

2. Check the Reachability Associations and Reachability Groups log pages to determine if there are two memory namespaces which do not share a reachability association, if none exist, then this test is not applicable.
3. Configure the NVMe Host to issue a Set Feature command to FID 16h to set Copy Descriptor Format 4h Enable to 1h.
4. Configure the NVMe Host to issue a Memory Copy command with Descriptor Format of 4h and NSID set to the first memory namespace and the SNSID in the Source Range Entry set to the second memory namespace which is not reachable to the first.

Observable Results:
1. Verify the Memory Copy command in step 4 completed with the status 'Namespace Not Reachable'.
2. Verify all other commands completed successfully.

Case 9: Memory Copy Command, SADDR & NBYTE, Not Dword Granular (FYI)

Test Procedure:
1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure for SLM commands (LID 05h, CSI 03h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable..
2. Configure the NVMe Host to issue a Set Feature command to FID 16h to set Copy Descriptor Format 4h Enable to 1h.
3. Configure the NVMe Host to issue a Memory Copy command with LEN=4h, SDADDR=8h, Descriptor Format=4h, NR=0h, and a Source Range Entry with the following values:
    a. SADDR=1h and NBYTE=4h.
4. Configure the NVMe Host to issue a Memory Copy command with LEN=4h, SDADDR=8h, Descriptor Format=4h, and NR=1h with two Source Range Descriptor Entries:
    a. An Entry with SADDR=0, NBYTE=1h; and
    b. An Entry with SADDR=4, NBYTE=3h.

Observable Results:
1. Verify both Memory Copy commands completed with the status 'Invalid Field'.
2. Verify all other commands completed successfully.

Case 10: Memory Copy Command, Format Descriptor 2h, SNSID (FYI)

Test Procedure:
1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure for SLM commands (LID 05h, CSI 3h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.
2. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 03h). Record the NOWG field.
3. Configure the NVMe Host to issue an Identify command to Identify Controller (CNS 1h) to determine if Copy Command Descriptor Format 2h is supported. If bit 2 of the Copy Descriptor Formats Supported field indicates a lack of support for descriptor format 2h, this test is not applicable.
4. Configure the NVMe Host to issue a Set Feature command to FID 16h to set Copy Descriptor Format Enable to 2h.
5. Configure the NVMe Host to issue three Memory Copy Commands each with a Descriptor Format of 2h, LEN=2^NOWG, SDADDR=0h, NR=0h, different invalid SNSID values:
    1. SNSID=0xFFFFFFFF;
    2. SNSID=0h; and

3. SNSID=The NSID of the memory namespace.

Observable Results:
1. Verify the Set Feature command was successful and Copy Command Descriptor Format 2h Enable is set to 1h.
2. Verify the Memory Copy commands in steps 5.a and 5.b returned with a status code of: 'Invalid Field'.
3. Verify the Memory Copy command in step 5.c returned with a status code of: 'Invalid Namespace or Format'.

Case 11: Memory Copy Command, Format Descriptor 3h, SNSID (FYI)

1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure for SLM commands (LID 05h, CSI 3h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.
2. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 03h). Record the NOWG field.
3. Configure the NVMe Host to issue an Identify command to Identify Controller (CNS 1h) data structure and determine if Copy Command Descriptor Format 3h is supported. If bit 3 of the Copy Descriptor Formats Supported field indicates a lack of support for descriptor format 3h, this test is not applicable.
4. Configure the NVMe Host to issue a Set Feature command to FID 16h to set Copy Descriptor Format Enable to 1h.
5. Configure the NVMe Host to issue three Memory Copy Commands each with a Descriptor Format of 2h, LEN=2^NOWG, SDADDR=0h, NR=0h, different invalid SNSID values:
    a. SNSID=0xFFFFFFFFF;
    b. SNSID=0h; and
    c. SNSID=The NSID of the memory namespace.

Observable Results:
1. Verify the Set Feature command was successful and Copy Command Descriptor Format h Enable is set to 1h.
2. Verify the Memory Copy commands in steps 5.a and 5.b returned with a status code of: 'Invalid Field'.
3. Verify the Memory Copy command in step 5.c returned with a status code of: 'Invalid Namespace or Format

Case 12: Memory Copy Command, Descriptor Format 2h, MCMSSRL, MCMCL, MCMSRC, LEN (FYI)

Test Procedure:
1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure for SLM commands (LID 05h, CSI 03h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.
2. Configure the NVMe Host to issue an Identify command to Identify Controller (CNS 1h) to determine if Copy Command Descriptor Format 2h is supported. If bit 2 of the Copy Descriptor Formats Supported field indicates a lack of support for descriptor format 2h, this test is not applicable.
3. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 3h). Record the MCMCL, MCMSSRL, and MCMSRC fields.
4. Determine if the DUT has an NVM namespace attached to the controller. If the DUT does not have an NVM namespace, issue a Namespace Management command to create one and then a Namespace Attachment command to attach it to the controller.
5. Determine the Reachability of the NVM namespace and the memory namespace. If there is no Reachability Association between the two namespaces this test is not applicable.

6. Configure the NVMe Host to issue a Set Feature command to FID 16h to set Copy Descriptor Format 2h Enable to 1h.
7. Configure the NVMe Host to issue 4 separate Memory Copy commands each specifying Descriptor Format 2h and SNSID of the NVM namespace:
    a. Memory Copy command with a single Source Range Entry with NBYTES=MCMSSRL+4.
    b. Memory Copy command with one or more Source Range Entries which cumulatively have NBYTES values that are equal to MCMCL+4.
    c. Memory Copy command with NR=MCMSRC+1.
    d. Memory Copy command with LEN=8h and two Source Range Entries with NBYTES=8h.

Observable Results:
1. Verify both Identify commands completed successfully.
2. Verify the Set and Get Feature commands were successful and Copy Command Descriptor Format 2h Enable is set to 1h.
3. Verify the Memory Copy command in step 7.a completed with the status ‘Command Size limit Exceeded’.
4. Verify the Memory Copy command in step 7.b completed with the status ‘Command Size limit Exceeded’.
5. Verify The Memory Copy command in step 7.c completed with the status ‘Command Size limit Exceeded’.
6. Verify the Memory Copy command in step 7.d completed with the status ‘Invalid Field’.

Case 13: Memory Copy Command, Descriptor Format 3h, MCMSSRL, MCMCL, MCMSRC, LEN (FYI)

Test Procedure:
1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure for SLM commands (LID 05h, CSI 03h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.
2. Configure the NVMe Host to issue an Identify command to Identify Controller (CNS 1h) to determine if Copy Command Descriptor Format 3h is supported. If bit 3 of the Copy Descriptor Formats Supported field indicates a lack of support for Descriptor Format 3h, this test is not applicable.
3. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 3h). Record the MCMCL, MCMSSRL, and MCMSRC fields.
4. Determine if the DUT has an NVM namespace attached to the controller. If the DUT does not have an NVM namespace, issue a Namespace Management command to create one and then a Namespace Attachment command to attach it to the controller.
5. Determine the Reachability of the NVM namespace and the memory namespace. If there is no Reachability Association between the two namespaces this test is not applicable.
6. Configure the NVMe Host to issue a Set Feature command to FID 16h to set Copy Descriptor Format 3h Enable to 1h.
7. Configure the NVMe Host to issue 4 separate Memory Copy commands each specifying Descriptor Format 3h and SNSID of the NVM namespace:
    a. Memory Copy command with a single Source Range Entry with NBYTES=MCMSSRL+4.
    b. Memory Copy command with one or more Source Range Entries which cumulatively have NBYTES values that are equal to MCMCL+4.
    c. Memory Copy command with NR=MCMSRC+1.
    d. Memory Copy command with LEN=8h and two Source Range Entries with NBYTES=8h.

Observable Results:
1. Verify both Identify commands completed successfully.
2. Verify the Set and Get Feature commands were successful and Copy Command Descriptor Format 3h Enable is set to 1h.
3. Verify the Memory Copy command in step 7.a completed with the status ‘Command Size Limit Exceeded’.
4. Verify the Memory Copy command in step 7.b completed with the status ‘Command Size Limit Exceeded’.

5. Verify The Memory Copy command in step 7.c completed with the status 'Command Size limit Exceeded'.
6. Verify the Memory Copy command in step 7.d completed with the status 'Invalid Field'.

Case 14: Memory Copy Command, Descriptor Format 4h, MCMSSRL, MCMCL, MCMSRC, LEN (FYI)

Test Procedure:

1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure for SLM commands (LID 05h, CSI 03h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.
2. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 3h). Record the MCMCL, MCMSSRL, and MCMSRC fields.
3. Configure the NVMe Host to issue a Set Feature command to FID 16h to set Copy Descriptor Format 4h Enable to 1h.
4. Configure the NVMe Host to issue 4 separate Memory Copy commands each specifying Descriptor Format 4h and SNSID of the memory namespace:
    a. Memory Copy command with a single Source Range Entry with NBYTES=MCMSSRL+4.
    b. Memory Copy command with one or more Source Range Entries which cumulatively have NBYTES values that are equal to MCMCL+4.
    c. Memory Copy command with NR=MCMSRC+1.
    d. Memory Copy command with LEN=8h and two Source Range Entries with NBYTES=8h.

Observable Results:

1. Verify the Identify command completed successfully.
2. Verify the Set and Get Feature commands were successful and Copy Command Descriptor Format 4h Enable is set to 1h.
3. Verify the Memory Copy command in step 4.a returned with a status code of 'Command Size Limit Exceeded'.
4. Verify the Memory Copy command in step 4.b returned with a status code of 'Command Size Limit Exceeded'.
5. Verify The Memory Copy command in step 4.c completed with the status 'Command Size limit Exceeded'
6. Verify the Memory Copy command in step 4.d returned with a status code of 'Invalid Field'.

Case 15: Memory Copy Command, Dword 0 (FYI)

Test Procedure:

1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure (LID 05h). If CSUPP is cleared to 0h for IOCS1 (Memory Copy Command) this test is not applicable.
2. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 3h). Record the NSZE, MCMCL, and MCMSRC fields.
3. Configure the NVMe Host to issue a Set Feature command to FID 16h to set Copy Descriptor Format 4h Enable to 1h.
4. Configure the NVMe Host to issue a Memory Write command with SB=0h and WL=MCMSRC * 4 to write all AAs to less than half of the memory namespace.
    a. If MCMSRC * 4 is greater than NSZE / 2 or MCMCL, instead specify WL=NSZE / 2.
    b. If MCMSRC * 4 and NSZE / 2 are both greater than MCML, instead specify WL=MCMCL.
5. Configure the NVMe Host to issue a Memory Copy command specifying Descriptor Format 4h, SNSID of the memory namespace and:
    a. A LEN value of the WL specified in the Memory Write command in step 4.
    b. A NR value equal to (LEN / 4) - 1, rounded down to the nearest integer.
    c. A SDADDR value of the WL specified in the Memory Write command in step 4

   d. A DPTR specifying a number of source range entries equal to the specified NR + 1 each specifying:
      i. SNSID = MNID of the current memory namespace
      ii. SADDR= LEN + n * 4
      iii. NBYTE = 4
6. Before the Completion Queue Entry for the Memory Copy command in step 5 is posted to the completion queue, Configure the NVMe Host to issue an Abort command to the controller specifying the CID of the Memory Copy command issued in step 5.
7. Configure the NVMe Host to issue a Memory Read command with SB and RL equal to the WL specified in the Memory Write command in step 4.

Observable Results:

1. If the Memory Copy command in step 5 returned a status of 'Command Abort Requested':
   a. Record the value returned in Command Dword 0 of the Memory Copy command in step 5 (DW0)
   b. Verify that the data buffer returned by the Memory Read command in step 7 reported a value not equal to AAh in at least one of the bytes between DW0 and DW0 + 4.
   c. Verify the Abort command in step 6 completed successfully and reported a value cleared to 0b in bit 1 of its returned Dword 0.
2. Verify all other commands completed successfully.

Possible Problems: The Controller may process the entirety of the Memory Copy command prior to the submission of the abort command. In this case, verify the Memory Copy command in step 5 completed successfully, the Memory Read command in step 7 returned a data buffer containing all 0xAA's, and the Abort command reports a value of 0b1 in bit 0 of its returned Dword 0.

Test 2.2 – Memory Write Command (FYI)

Purpose: To verify that an NVMe Controller is able to successfully write data to specified namespaces using the Memory Write command.

References:

[1] NVM Express Subsystem Local Memory Command Set Specification 1.0 : 3.2.4

Resource Requirements:

Tools capable of monitoring and decoding traffic on the NVMe interface.

Last Modification: July 8, 2024

Discussion: The Memory Write command is used to write data to specific namespaces. This command is required to be Dword granular and Dword aligned. The command writes data to the NVM subsystem local memory only, and cannot access the logical blocks in storage namespaces.

The Memory Write command points to the data to be written, contains the Starting Byte (SB) address to write to, and the Write Length (WL) of the length of the write in bytes.

Test Setup: See Appendix A.

Case 1: Memory Write Command (FYI)

Test Procedure:

1. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 03h). Record the NOWG field.
2. Configure the NVMe Host to issue a Memory Write command with SB=0h and WL=2^NOWG to write all AAs to the memory namespace.
3. Configure the NVMe Host to issue a Memory Read command with SB=0h and RL=2^NOWG to read the byte range previously written to.

Observable Results:

1. Verify all commands completed successfully.
2. The data pattern returned by the Read command from step 3 was all AAs.

Case 2: Memory Write Command, Not Dword Aligned (FYI)

Test Procedure:

1. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set Specific Identify Namespace data structure (CNS 05h, CSI 03h). Record the NOWG field.
2. Configure the NVMe Host to issue a Memory Write command with SB=0h and WL=1.
3. Configure the NVMe Host to issue a Memory Write command with SB=1h and WL=(2^NOWG).

Observable Results:

1. Verify the Identify command completed successfully.
2. Verify both Memory Write commands returned with a status code of ‘Invalid Field in Command’.

Test 2.3 – Memory Read Command (FYI)

Purpose: To verify that an NVMe Controller can successfully read data to specified namespaces using the Memory Read command.

References:

[1] NVM Express Subsystem Local Memory Command Set Specification 1.0 : 3.2.3

Resource Requirements:

Tools capable of monitoring and decoding traffic on the NVMe interface.

Last Modification: July 8, 2024

Discussion: The Memory Read command is used to write data to specific namespaces. This command is required to be Dword granular and Dword aligned. The command reads data to the NVM subsystem local memory only and cannot access the logical blocks in storage namespaces.

The Memory Read command points to the data to be read, contains the Starting Byte (SB) address to read from, and the Read Length (RL) of the length of the write in bytes.

Test Setup: See Appendix A.

Case 1: Memory Read Command (FYI)

Test Procedure:

1. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 03h). Record the NOWG field.
2. Configure the NVMe Host to issue a Memory Write command with SB=0h and WL=2^NOWG to write all FFs to the memory namespace.
3. Configure the NVMe Host to issue a Memory Read command with SB=0h and RL=2^NOWG to read the byte range previously written to.

Observable Results:

1. Verify all commands completed successfully.
2. The data pattern returned by the Read command from step 3 was all FFs

Case 2: Memory Read Command, Not Dword Aligned (FYI)

Test Procedure:

1. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set Specific Identify Namespace data structure (CNS 05h, CSI 03h). Record the NOWG field.
2. Configure the NVMe Host to issue a Memory Read command with SB=0h and RL=1.
3. Configure the NVMe Host to issue a Memory Read command with SB=1h and RL=(2^NOWG).

Observable Results:

1. Verify the Identify command completed successfully.
2. Verify both Memory Read commands returned with a status code of ‘Invalid Field in Command’.

Test 2.4 – Memory Fill Command (FYI)

Purpose: To verify that an NVMe Controller can successfully fill a memory range with zeros using the Memory Fill Command.

References:

[1] NVM Express Subsystem Local Memory Command Set Specification 1.0 : 3.2.3

Resource Requirements:

Tools capable of monitoring and decoding traffic on the NVMe interface.

Last Modification: July 8, 2024

Discussion: The Memory Fill command is used to write zeros to specific namespaces. This command is required to be Dword granular and Dword aligned.

The Memory Fill command does not require the Data Pointer field, but still contains the Starting Byte (SB) address to write to, and the Fill Length (FL) of the length of the write in bytes.

Case 1: Memory Fill Command (FYI)

Test Procedure:

1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure (LID 05h) for CSI=03h. If CSUPP is cleared to 0h for IOCS0 (Memory Fill Command) this test is not applicable.
2. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 03h). Record the NOWG field.
3. Configure the NVMe Host to issue a Memory Write command with SB=0h and WL=2^NOWG to write all FFs to the memory namespace.
4. Configure the NVMe Host to issue a Memory Read command with SB=0h and RL=2^NOWG to read the byte range previously written to.
5. Configure the NVMe Host to issue a Memory Fill command with SB=0h and FL=2^NOWG to fill zeros over the data pattern written.
6. Configure the NVMe Host to issue a Memory Read command with SB=0h and RL=2^NOWG to read the byte range previously filled.

Observable Results:

1. Verify all commands completed successfully.
2. The data pattern returned by the Read command from step 4 was all FFs.
3. The data pattern returned by the Read command from step 6 was all 00s.

Case 2: Memory Fill Command, Not Dword Aligned (FYI)

Test Procedure:

1. Configure the NVMe Host to issue a Get Log command to obtain the Command Supported and Effects log page data structure (LID 05h) for CSI=03h. If CSUPP is cleared to 0h for IOCS0 (Memory Fill Command) this test is not applicable.

2. Configure the NVMe Host to issue an Identify command to obtain the I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 03h). Record the NOWG field.
3. Configure the NVMe Host to issue a Memory Fill command with SB=0h and FL=1.
4. Configure the NVMe Host to issue a Memory Fill command with SB=1h and FL=(2^NOWG).

Observable Results:

1. Verify both Memory Fill commands returned with a status code of 'Invalid Field in Command'.
2. Verify all other commands completed successfully.

Group 3: Namespace Management Commands

Test 3.1 – Namespace Attachment (FYI)

Purpose: To verify that an NVMe Controller that contains memory namespaces operates in the same way that the NVMe Express Base specification defines, and can be attached and detached.

References:

[1] NVM Express Subsystem Local Memory Command Set Specification 1.0 : 3.2.3
[2] NVM Express Base Specification 2.0d : 5.22

Resource Requirements:

Tools capable of monitoring and decoding traffic on the NVMe interface.

Last Modification: July 8, 2024

Discussion: Memory namespaces do not support

Case 1: Namespace Attachment (FYI)

Test Procedure:

1. Configure the NVMe Host to issue an Identify command with (CNS 07h, CSI 03h). Record the NSID value for each namespace if reported.
2. For each namespace which reports a CSI value of 3h:
    a. Configure the NVMe Host to issue a Namespace Attachment command with SEL=0h to detach the namespace from an attached controller.
    b. Configure the NVMe Host to issue an Namespace Attachment command with SEL=1h to attach the namespace back to the controller.

Observable Results:

1. Verify all commands completed successfully.

Test 3.2 – Namespace Management (FYI)

Purpose: To verify that an NVMe Controller that contains memory namespaces does not allow for the use of the namespace management command on the memory namespace.

References:

[1] NVM Express Subsystem Local Memory Command Set Specification 1.0 : 4.1.8

Resource Requirements:

Tools capable of monitoring and decoding traffic on the NVMe interface.

Last Modification: July 8, 2024

Discussion: Memory namespaces do not support namespace management. Memory namespaces cannot be deleted or created.

Test Setup: See Appendix A.

Case 2: Namespace Management (FYI)

Test Procedure:

1. Configure the NVMe Host to issue a Namespace Management command with SEL=0h and CSI=3h to attempt to create a namespace.
2. Configure the NVMe Host to issue a Namespace Management command with SEL=1h to attempt to delete a namespace with a CSI value of 3h.

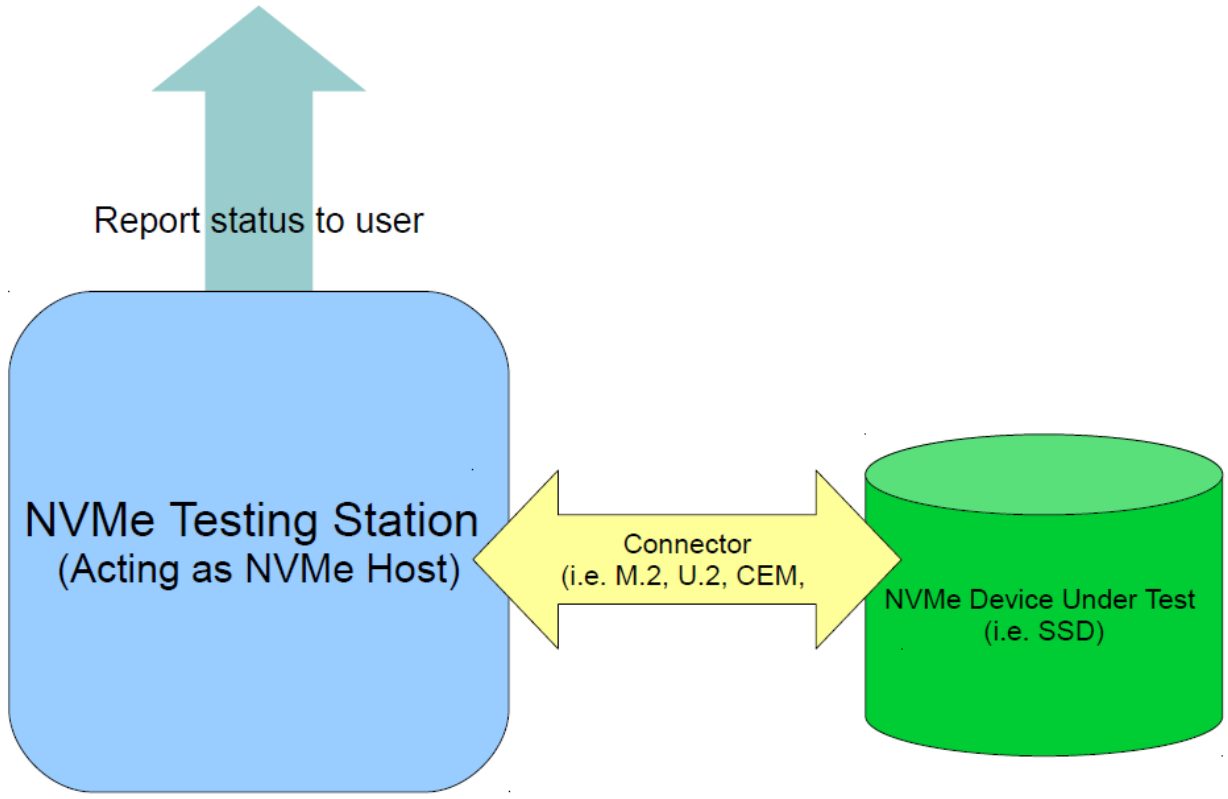
Observable Results:

1. Verify both Namespace Management commands returned with the status code of Invalid Field in Command.

Appendix A: DEFAULT TEST SETUP

Except where otherwise specified, all tests will require the DUT to have one of the following default physical configuration at the beginning of each test case:

Test Setup for NVMe Device:



Appendix B: NOTES ON TEST PROCEDURES

There are scenarios where in test procedures it is desirable to leave certain aspects of the testing procedure as general as possible. In these cases, the steps in the described test procedure may use placeholder values, or may intentionally use non–specific terminology, and the final determination of interpretation or choice of values is left to the discretion of the test technician. The following is an attempt to capture and describe all such instances used throughout the procedures.

Ports on Testing Station and Device Under Test	In general, any PCIe Port on the Testing Station or Device Under Test may be used as an interface with a test station or interoperability partner. There is *assumed* to be no difference in behavior, with respect to the protocols involved in this test suite, between any two PCIe ports on the Testing Station or Device Under Test. Hence, actual ports used may be chosen for convenience. However, it is recommended that the PCIe port used in the test configuration is recorded by the test technician.
Use of "various"	To maintain generality, some steps will specify that "various other values" (or the like) should be used in place of a given parameter. Ideally, all possible values would be tested in this case. However, limits on available time may constrain the ability of the test technician to attempt this. Given this, a subset of the set of applicable values must generally be used. When deciding how many values should be used, it should be noted that the more values that are tested, the greater the confidence of the results obtained (although there is a diminishing return on this). When deciding which specific values to use, it is generally recommended to choose them at pseudo–randomly yet deterministically. However, if there exist subsets of the applicable values with special significance, values from each subset should be attempted.

Appendix C: TEST TOOLS

The Tests described in this document can be performed using available IOL INTERACT NVMe Test Software available from UNH–IOL.

If using the PC Edition of the IOL INTERACT NVMe Test Software, UNH-IOL recommends using v21.0 or higher of the IOL INTERACT NVMe Test Software. This software is available via https://www.iol.unh.edu/solutions/test-tools/interact

Appendix D: NVME INTEGRATORS LIST REQUIREMENTS

Purpose: To provide guidance on what tests are required for NVMe Integrators List Qualification

References:

[1] NVMe Integrators List Policy Document

Resource Requirements:

NVMe Host Platform and Device.

Last Modification: April 2, 2019

Discussion: Each Test defined in this document is defined as being Mandatory (M), FYI, or In Progress (IP). This primary designation is shown in the title of the test case and is understood to apply to PCIe based products. An additional designation is provided if a test is applicable to NVMe-oF products (OF). Tests that are designated as being applicable to NVMe-oF Products are understood to inherit the primary designation of the test (i.e. M, FYI, IP), unless an additional designation is specified. The following examples are provided:

Test 1.1 Example Name (M)– Test is mandatory for all PCIe based products and does not apply to NVMe-oF products.
Test 2.1 Example Name (M, OF)– Test is mandatory for all products, including NVMe-oF products.
Test 3.1 Example Name (M, OF-IP)- Test is mandatory for all PCIe based products, and test is currently On Progress for NVMe-oF products.

NVMe protocol testing is independent of the transport used. Conformance tests described in this document may be performed at any link speed, width, or transport type that the NVMe product under test supports.

If a Test is designated as Mandatory, a product must pass this test in order to qualify for the NVMe Integrators List. For tests that deal with features defined as optional in the NVMe specification, a check is performed at the beginning of the test to determine if the optional feature is supported or not. If the optional feature is not supported the test is marked as 'Not Applicable' and does not impact qualification for the Integrators List.

If a Test is designated as FYI, a device does not need to pass this test in order to qualify for the NVMe Integrators List. Tests designated as FYI may become Mandatory tests in the future.

If a Test is designated as In Progress, a device does not need to pass this test in order to qualify for the NVMe Integrators List. These test cases are still under development. Tests designated as In Progress may become Mandatory tests in the future.

Any Test may have a Case within it with a different designation as the Test itself (i.e. a Mandatory test may include FYI cases). In this case, only the Mandatory Cases are required for NVMe Integrators List qualification.