Personal Area Networking Profile (PAN)

***Bluetooth*® Test Suite**

- **Revision:** PAN.TS.p12
- **Revision Date:** 2024-07-01
- **Prepared By:** BTI
- **Published during TCRL:** TCRL.2024-1

Contents

1 Scope

This Bluetooth document contains the Test Suite Structure (TSS) and test cases to test the implementation of the Bluetooth Personal Area Networking Profile (PAN) with the objective to provide a high probability of air interface interoperability between the tested implementation and other manufacturers' Bluetooth devices.

2 References, definitions, and abbreviations

2.1 References

This document incorporates provisions from other publications by dated or undated reference. These references are cited at the appropriate places in the text, and the publications are listed hereinafter. Additional definitions and abbreviations can be found in [1], [2], and [13].

[1] Bluetooth Network Encapsulation Protocol (BNEP) Specification, Version 1.0

[2] Bluetooth Core Specification, Version 1.2 or later

[3] Bluetooth SIG, “Bluetooth Assigned Number”, https://www.bluetooth.com/specifications/assigned-numbers

[4] Internet Engineering Task Force, “Dynamic Configuration of IPv4 Link-Local Addresses”, RFC3927, May 2005

[5] Personal Area Networking Profile (PAN) Specification, Version 1.0

[6] Internet Engineering Task Force, “IPv6 Stateless Address Autoconfiguration”, RFC2462, December 1998

[7] Internet Engineering Task Force, “Dynamic Host Configuration Protocol”, RFC2131, March 1997

[8] Internet Engineering Task Force, “DHCP Options and BOOTP Vendor Extensions”, RFC2132, March 1997

[9] Internet Engineering Task Force, “DOMAIN NAMES - CONCEPTS AND FACILITIES”, RFC1034, November 1987

[10] Internet Engineering Task Force, “DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION”, RFC1035, November 1987

[11] Internet Engineering Task Force, “INTERNET CONTROL MESSAGE PROTOCOL”, RFC792, September 1981

[12] Internet Engineering Task Force, “INTERNET CONTROL MESSAGE PROTOCOL (ICMPv6)”, RFC2463, December 1998

[13] Test Strategy and Terminology Overview

[14] ICS Proforma for Personal Area Networking (PAN)

[15] SDP Test Suite, SDP.TS

[16] IXIT Proforma for Personal Area Networking Profile (PAN)

[17] Appropriate Language Mapping Tables document

2.2 Definitions

In this Bluetooth document, the definitions from [1], [2], and [13] apply.

Certain terms that were identified as inappropriate have been replaced. For a list of the original terms and their replacement terms, see the Appropriate Language Mapping Tables document [17].

2.3 Acronyms and abbreviations

In this Bluetooth document, the definitions, acronyms, and abbreviations from [1], [2], and [13] apply.

3 Test Suite Structure (TSS)

3.1 Test Strategy

The PAN profile test strategy incorporates conformance testing of the Profile and the Bluetooth Networking Encapsulation Protocol (BNEP).

In these tests, two representative topologies are required.

- Topology 1 represents an ad-hoc network without networking infrastructure such as DHCP and DNS servers.
- Topology 2 consists of an access point that spans the piconet and another network.

While the two topologies do not represent the complete set of topologies that are possible in a PAN environment, they are representative of the most typical topologies and suitable for testing.

3.1.1 Topology 1, Ad-Hoc Mode

In the Ad-hoc Mode topology, one device performs the GN role and all of the other devices are performing the PANU role.

This topology consists of an ad-hoc network, as shown in Figure 3.1.

This topology requires at least the device providing the GN service and a single client PANU as shown in Example B. Some tests require multiple clients as shown in Example A.

The IUT may be the GN or one of the PANUs in the tests that rely on this topology.

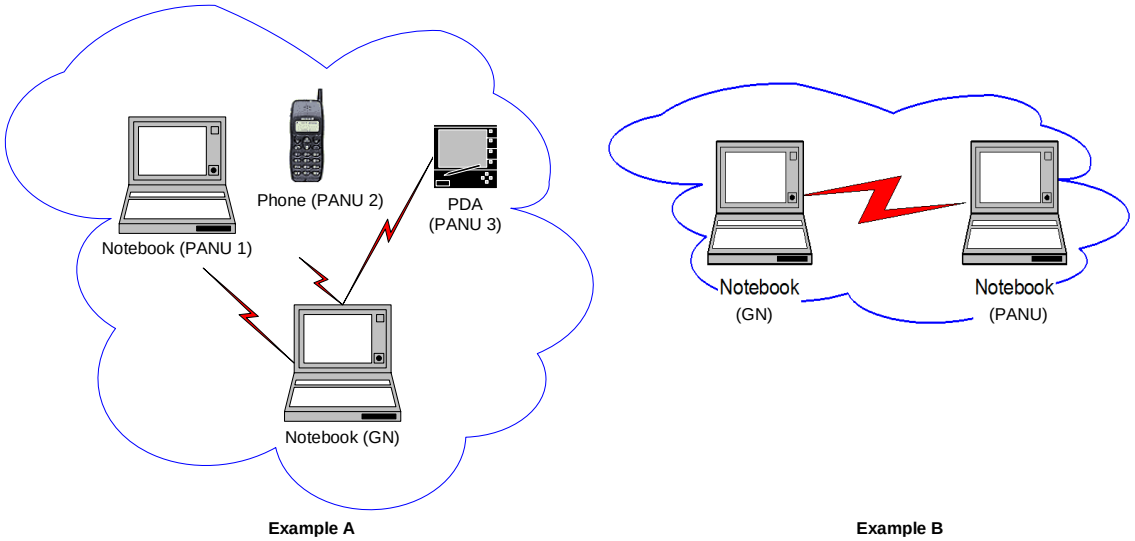

Figure 3.1: Sample Ad-Hoc Network Topologies; Example A: Multiple PANUs, Example B: Single PANU

3.1.2 Topology 2, Infrastructure Mode

In the Infrastructure Mode topology, one device performs the NAP role and all of the other devices are performing the PANU role. This test topology includes network infrastructure.

Different infrastructure mechanisms are possible (such as GSM, CDMA, GPRS, Ethernet, WAN, WLAN, LAN, etc.) and may be used in the test topology. An example of this topology is shown in Figure 3.2.

If the IUT is acting as a NAP, the Network Terminal, as the Upper Tester, may be on a physically separate device; if that is the case, then network connectivity between the two is needed for carrying out tests in this Test Suite.

In Example A, multiple client PANUs are shown. Some tests may be run with a single client, as shown in Example B.

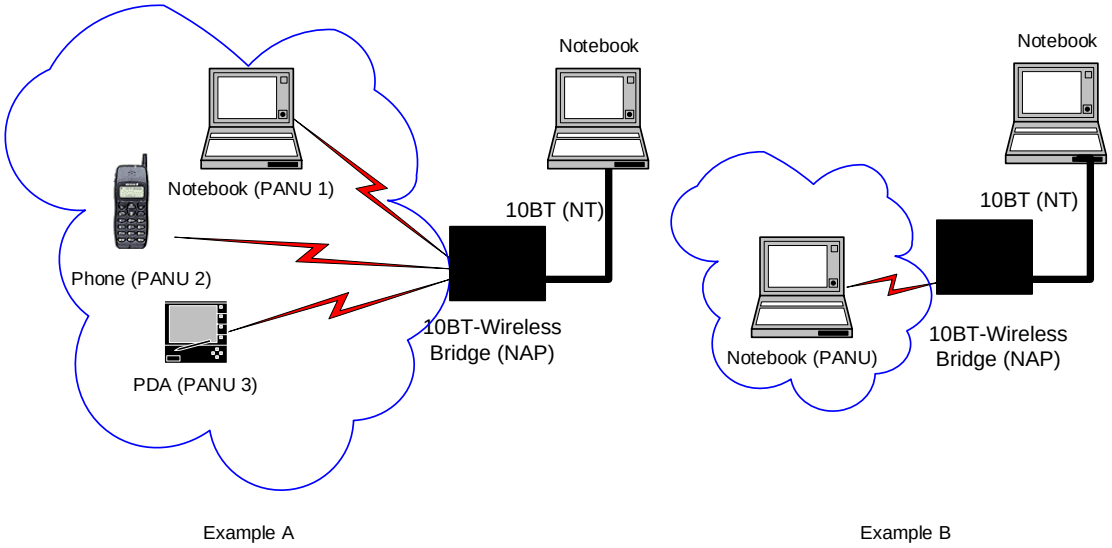

Figure 3.2: Sample Network Infrastructure Topologies; Example A: Multiple PANUs, Example B: Single PANU

4 Test cases (TC)

4.1 Introduction

4.1.1 Test case identification conventions

Test cases are assigned unique identifiers per the conventions in [13]. The convention used here is:
<spec abbreviation>/<IUT role>/<class>/**<feat>**/<func>/<subfunc>**/**<cap>**/<xx>-<nn>-<y>**.

Additionally, testing of this specification includes tests from the SDP Test Suite [15] referred to as Generic SDP Integrated Tests (GSIT); when used, the test cases in GSIT are referred to through a TCID string using the following convention:
<spec abbreviation>/<IUT role>/<GSIT test group>/<GSIT class>/<xx>-<nn>-<y>.

Identifier Abbreviation	**Spec Identifier <spec abbreviation>**
PAN	Personal Area Network
Identifier Abbreviation	**Role Identifier <IUT role>**
GN	Group Ad-hoc Network role
NAP	Network Access Point role
PANU	PAN User role
Identifier Abbreviation	**Reference Identifier <GSIT test group>**
CGSIT	Client Generic SDP Integrated Tests
SGSIT	Server Generic SDP Integrated Tests
Identifier Abbreviation	**Reference Identifier <GSIT class>**
ATTR	Attribute
OFFS	Attribute ID Offset String
SERR	Service Record
SFC	SDP Future Compatibility
Identifier Abbreviation	**Class Identifier <class>**
BNEP	Bluetooth Network Encapsulation Protocol
BRIDGE	Packet forwarding
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
IP	Internet Protocol
MISC	Miscellaneous
Identifier Abbreviation	**Feature Identifier <feat>**
APP	Application test
Autonet	Autonet
BROADCAST	Data Broadcast
DHCP	Dynamic Host Configuration Protocol server
DNS	Domain Name Service server
EXTENSION	Extension headers
FILTER	Network Protocol Filter
FORWARD	Data forward

Identifier Abbreviation	Feature Identifier <feat>
LLMNR	Linklocal Multicast Name Resolution
MULTICAST	Data Multicast
ROLE	PAN role-specific tests
RX	Receive
TX	Transmit
UUID	Universally Unique Identifier

Table 4.1: PAN TC feature naming convention

4.1.2 Conformance

When conformance is claimed for a particular specification, all capabilities are to be supported in the specified manner. The mandated tests from this Test Suite depend on the capabilities to which conformance is claimed.

The Bluetooth Qualification Program may employ tests to verify implementation robustness. The level of implementation robustness that is verified varies from one specification to another and may be revised for cause based on interoperability issues found in the market.

Such tests may verify:

- That claimed capabilities may be used in any order and any number of repetitions not excluded by the specification
- That capabilities enabled by the implementations are sustained over durations expected by the use case
- That the implementation gracefully handles any quantity of data expected by the use case
- That in cases where more than one valid interpretation of the specification exists, the implementation complies with at least one interpretation and gracefully handles other interpretations
- That the implementation is immune to attempted security exploits

A single execution of each of the required tests is required to constitute a Pass verdict. However, it is noted that to provide a foundation for interoperability, it is necessary that a qualified implementation consistently and repeatedly pass any of the applicable tests.

In any case, where a member finds an issue with the test plan generated by the Bluetooth SIG qualification tool, with the test case as described in the Test Suite, or with the test system utilized, the member is required to notify the responsible party via an erratum request such that the issue may be addressed.

4.1.3 Pass/Fail verdict conventions

Each test case has an Expected Outcome section. The IUT is granted the Pass verdict when all the detailed pass criteria conditions within the Expected Outcome section are met.

The convention in this Test Suite is that, unless there is a specific set of fail conditions outlined in the test case, the IUT fails the test case as soon as one of the pass criteria conditions cannot be met. If this occurs, then the outcome of the test is a Fail verdict.

4.2 Generic SDP Integrated Tests

4.2.1 Server Generic SDP Integrated Tests

4.2.1.1 Personal Area Network Profile – NAP

Execute the Generic SDP Integrated Tests defined in Section 6.3, Server test procedures (SGSIT), in [15] using Table 4.2 below as input:

TCID	Reference	Attribute ID name	Attribute ID definition source (Universal, Profile)	Value/secondary value	Attribute presence (Present/Present for [role], Optionally present, TCMT defined)
PAN/NAP/SGSIT/SERR/BV-01-C [Service record GSIT – PAN NAP]	[5] 8.1.1	ServiceClassIDList	Universal	"NAP" (UUID)	Present for NAP
PAN/NAP/SGSIT/ATTR/BV-01-C [Attribute GSIT – Protocol Descriptor List]	[5] 8.1.1	ProtocolDescriptorList	Universal	"L2CAP" (UUID): PSM – skip (Uint16), "BNEP" (UUID): Version – "0x0100" (Uint16), Supported Network Packet Type List – skip, (Data Element Sequence)	Present for NAP
PAN/NAP/SGSIT/ATTR/BV-02-C [Attribute GSIT – Language Base Attribute ID List]	[5] 8.1.1	LanguageBaseAttributeIDList	Universal	skip (Data Element Sequence)	Present for NAP
PAN/NAP/SGSIT/ATTR/BV-03-C [Attribute GSIT – Service Availability]	[5] 8.1.1	ServiceAvailability	Universal	skip (Uint8)	Optionally present
PAN/NAP/SGSIT/ATTR/BV-04-C [Attribute GSIT – Bluetooth Profile Descriptor List]	[5] 8.1.1	BluetoothProfileDescriptorList	Universal	"NAP" (UUID): Version – "0x0100" (Uint16)	Present for NAP
PAN/NAP/SGSIT/ATTR/BV-05-C [Attribute GSIT – Security Description]	[5] 8.1.1	Security Description	Profile	skip (Uint16)	Present for NAP
PAN/NAP/SGSIT/ATTR/BV-06-C [Attribute GSIT – NetAccessType]	[5] 8.1.1	NetAccessType	Profile	skip (Uint16)	Present for NAP
PAN/NAP/SGSIT/ATTR/BV-07-C [Attribute GSIT – MaxNetAccessRate]	[5] 8.1.1	MaxNetAccessRate	Profile	skip (Uint32)	Present for NAP
PAN/NAP/SGSIT/ATTR/BV-08-C [Attribute GSIT – IPv4Subnet]	[5] 8.1.1	IPv4Subnet	Profile	skip (String)	Optionally present
PAN/NAP/SGSIT/ATTR/BV-09-C [Attribute GSIT – IPv6Subnet]	[5] 8.1.1	IPv6Subnet	Profile	skip (String)	Optionally present

Table 4.2: Input for the Personal Area Networking Profile NAP SGSIT SDP test procedure

4.2.1.2 Personal Area Networking Profile – GN

Execute the Generic SDP Integrated Tests defined in Section 6.3, Server test procedures (SGSIT), in [15] using Table 4.3 below as input:

TCID	Reference	Attribute ID name	Attribute ID definition source (Universal, Profile)	Value/secondary value	Attribute presence (Present/Present for [role], Optionally present, TCMT defined)
PAN/GN/SGSIT/SERR/BV-02-C [Service record GSIT – PAN GN]	[5] 8.1.2	ServiceClassIDList	Universal	"GN" (UUID)	Present for GN
PAN/GN/SGSIT/ATTR/BV-10-C [Attribute GSIT – Protocol Descriptor List]	[5] 8.1.2	ProtocolDescriptorList	Universal	"L2CAP" (UUID): PSM – skip (Uint16), "BNEP" (UUID): Version – "0x0100" (Uint16), Supported Network Packet Type List – skip (Data Element Sequence)	Present for GN
PAN/GN/SGSIT/ATTR/BV-11-C [Attribute GSIT – Language Base Attribute ID List]	[5] 8.1.2	LanguageBaseAttributeIDList	Universal	skip (Data Element Sequence)	Present for GN
PAN/GN/SGSIT/ATTR/BV-12-C [Attribute GSIT – Service Availability]	[5] 8.1.2	ServiceAvailability	Universal	skip (Uint8)	Optionally present
PAN/GN/SGSIT/ATTR/BV-13-C [Attribute GSIT – Bluetooth Profile Descriptor List]	[5] 8.1.2	BluetoothProfileDescriptorList	Universal	"GN" (UUID): Version – "0x0100" (Uint16)	Present for GN
PAN/GN/SGSIT/ATTR/BV-14-C [Attribute GSIT – Security Description]	[5] 8.1.2	Security Description	Profile	skip (Uint16)	Present for GN
PAN/GN/SGSIT/ATTR/BV-15-C [Attribute GSIT – IPv4Subnet]	[5] 8.1.2	IPv4Subnet	Profile	skip (String)	Optionally present
PAN/GN/SGSIT/ATTR/BV-16-C [Attribute GSIT – IPv6Subnet]	[5] 8.1.2	IPv6Subnet	Profile	skip (String)	Optionally present

Table 4.3: Input for the Personal Area Networking Profile GN SGSIT SDP test procedure

4.2.1.3 Personal Area Networking Profile – PANU

Execute the Generic SDP Integrated Tests defined in Section 6.3, Server test procedures (SGSIT), in [15] using Table 4.4 below as input:

TCID	Reference	Attribute ID name	Attribute ID definition source (Universal, Profile)	Value/secondary value	Attribute presence (Present/Present for [role], Optionally present, TCMT defined)
PAN/PANU/SGSIT/SERR/BV-03-C [Service record GSIT – PAN PANU]	[5] 8.1.3	ServiceClassIDList	Universal	PANU	Present for PANU
PAN/PANU/SGSIT/ATTR/BV-17-C [Attribute GSIT – Protocol Descriptor List]	[5] 8.1.3	ProtocolDescriptorList	Universal	"L2CAP" (UUID): PSM – skip (Uint16), "BNEP" (UUID): Version – "0x0100" (Uint16), Supported Network Packet Type List – skip (Data Element Sequence)	Present for PANU
PAN/PANU/SGSIT/ATTR/BV-18-C [Attribute GSIT – Language Base Attribute ID List]	[5] 8.1.3	LanguageBaseAttributeIDList	Universal	skip (Data Element Sequence)	Present for PANU
PAN/PANU/SGSIT/ATTR/BV-19-C [Attribute GSIT – Service Availability]	[5] 8.1.3	ServiceAvailability	Universal	skip (Uint8)	Optionally present
PAN/PANU/SGSIT/ATTR/BV-20-C [Attribute GSIT – Bluetooth Profile Descriptor List]	[5] 8.1.3	BluetoothProfileDescriptorList	Universal	"PANU" (UUID): Version – "0x0100" (Uint16)	Present for PANU
PAN/PANU/SGSIT/ATTR/BV-21-C [Attribute GSIT – Security Description]	[5] 8.1.3	Security Description	Profile	skip (Uint16)	Present for PANU

Table 4.4: Input for the Personal Area Networking Profile PANU SGSIT SDP test procedure

4.2.1.4 Personal Area Networking Profile – Attribute ID Offset String tests

Execute the Generic SDP Integrated Tests defined in Section 6.3, Server test procedures (SGSIT), in [15] using Table 4.5 below as input:

TCID	Reference	ServiceSearchPattern	Attribute ID name	Attribute ID Offset	Attribute presence (Present/Present for [role], Optionally present, TCMT defined)
PAN/NAP/SGSIT/OFFS/BV-01-C [Attribute ID Offset String GSIT – Service Name]	[5] 8.1.1	NAP	ServiceName	0x0000	Present for NAP

TCID	Reference	ServiceSearchPattern	Attribute ID name	Attribute ID Offset	Attribute presence (Present/Present for [role], Optionally present, TCMT defined)
PAN/NAP/SGSIT/OFFS/BV-02-C [Attribute ID Offset String GSIT – Service Description]	[5] 8.1.1	NAP	ServiceDescription	0x0001	Present for NAP
PAN/GN/SGSIT/OFFS/BV-03-C [Attribute ID Offset String GSIT – Service Name]	[5] 8.1.2	GN	ServiceName	0x0000	Present for GN
PAN/GN/SGSIT/OFFS/BV-04-C [Attribute ID Offset String GSIT – Service Description]	[5] 8.1.2	GN	ServiceDescription	0x0001	Present for GN
PAN/PANU/SGSIT/OFFS/BV-05-C [Attribute ID Offset String GSIT – Service Name]	[5] 8.1.3	PANU	ServiceName	0x0000	Present for PANU
PAN/PANU/SGSIT/OFFS/BV-06-C [Attribute ID Offset String GSIT – Service Description]	[5] 8.1.3	PANU	ServiceDescription	0x0001	Present for PANU

Table 4.5: Input for the Personal Area Networking Profile SGSIT Attribute ID Offset String tests

4.2.2 Client Generic SDP Integrated Tests

Execute the Generic SDP Future Compatibility Tests defined in Section 6.4, Client test procedures (CGSIT), in [15] using Table 4.6 below as input:

TCID	Reference	Service Record Service Class UUID description	Lower Tester SDP record initial conditions
PAN/PANU/CGSIT/SFC/BV-01-C [SDP Future Compatibility – IUT is PAN PANU]	[5] 8	NAP or GN	Lower Tester exposes either a PAN NAP, GN, or PANU SDP record (peer role is specified by IXIT [16]). The version in the Bluetooth Profile Descriptor List is greater than the most recently adopted version. If the peer role is PAN NAP, all bits are set in the NetAccessType attribute, including Reserved bits.

Table 4.6: Input for the Client CGSIT SDP future compatibility tests

4.3 PAN Broadcast and Multicast test cases

PAN/NAP/BNEP/BROADCAST-0/BV-01-C [Broadcasting from a Bluetooth Network to Other Networks]

- Test Purpose

  Verify the ability of the IUT (NAP) to handle broadcasting packets from the Lower Tester (PANU) to the network using different compression formats.

- Reference

  [5] 5.1, 5.2, 5.7, 6.1.1

  [1] 2.5, 2.9

- Initial Condition

  - Both the IUT (NAP) and the Lower Tester (PANU) are initialized.
  - Initialization may require a PIN if there are any security restrictions already in place on the Bluetooth link.
  - The IUT has an open and configured BNEP connection with the Lower Tester. The IUT is the initiator and the Lower Tester is the acceptor in the Bluetooth connection.

- Test Procedure

  1. The Lower Tester transmits a BNEP packet to the IUT as decribed in Table 4.7.
  2. The IUT removes the BNEP header, adds that Ethernet header, and transmits the packet to the Upper Tester.

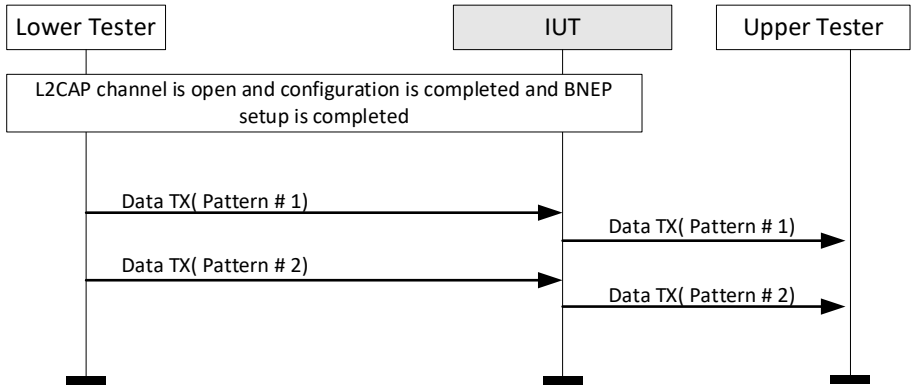

Figure 4.1: Broadcasting from a Bluetooth Network to Other Networks

In step 1, the Lower Tester sends two test patterns as described below:

Pattern #1	**Value**
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	IEEE broadcast address (FF:FF:FF:FF:FF:FF)
Source Address (48 bits)	Lower Tester (PANU)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored

Pattern #1	Value
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Pattern #2	Value
L2CAP Length	0x5E5 (BNEP packet length within the L2CAP packet)
BNEP Type	0x04 (BNEP_COMPRESSED_ETHERNET_DEST_ONLY)
Destination Address (48 bits)	IEEE broadcast address (FF:FF:FF:FF:FF:FF)
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.7: Broadcasting from a Bluetooth Network to Other Networks – Data TX patterns

- Expected Outcome

  Pass verdict

  The Upper Tester receives Ethernet broadcast packets as described below in Table 4.8.

  Both test patterns result in identical Ethernet packets on the Upper Tester.

Pattern #n	Value
Destination Address (48 bits)	IEEE broadcast address (FF:FF:FF:FF:FF:FF)
Source Address (48 bits)	Lower Tester (PANU)'s Bluetooth Address'
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.8: Broadcasting from a Bluetooth Network to Other Networks – Pass criteria

PAN/NAP/BNEP/BROADCAST-1/BV-02-C [Broadcasting from Another Network to a Bluetooth Network]

- Test Purpose

  Verify the use of the BNEP_GENERAL_ETHERNET (0x00) packet when broadcasting packets from the Upper Tester (Network Terminal) through an IUT (NAP) to the Lower Tester (PANU).

- Reference

  [5] 5.1, 5.2, 5.7, 6.1.1

  [1] 2.5

- Initial Condition

  - Both the IUT (NAP) and the Lower Tester (PANU) are initialized.
  - Initialization may require a PIN if there are any security restrictions already in place on the Bluetooth link.

- The IUT has an open and configured BNEP connection with the Lower Tester.

- Test Procedure

1. The Upper Tester (Network Terminal) sends an Ethernet broadcast packet; the test pattern is as described in Table 4.9.
2. The IUT forwards the broadcast Ethernet packet to the Lower Tester as a BNEP packet.

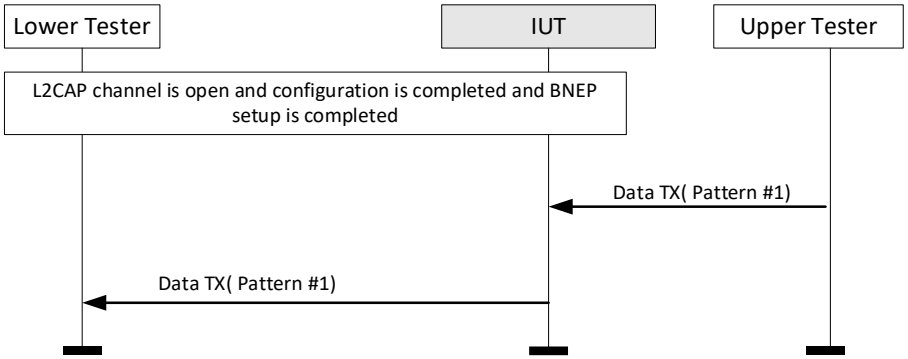

Figure 4.2: Broadcasting from Another Network to a Bluetooth Network

In step 1, the Upper Tester sends a test pattern as described below:

Pattern #1	**Value**
Destination Address (48 bits)	IEEE broadcast address (FF:FF:FF:FF:FF:FF)
Source Address (48 bits)	Upper Tester (Network Terminal's Ethernet Address)
Network Protocol Type	0x0800, or 0x86DD, or 0xXXXX
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.9: Broadcasting from Another Network to a Bluetooth Network – Data TX pattern

- Expected Outcome

Pass verdict

The IUT sends the packet described below to the Lower Tester:

Pattern #1	**Value**
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	IEEE broadcast address (FF:FF:FF:FF:FF:FF)
Source Address (48 bits)	Upper Tester (Network Terminal's Ethernet Address)
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.10: Broadcasting from Another Network to a Bluetooth Network – Pass criteria

PAN/NAP/BNEP/MULTICAST-0/BV-03-C [Multicasting from a Bluetooth Network to Other Networks]

- Test Purpose

  Verify the ability of the IUT (NAP) to handle multicasting packets from the Lower Tester (PANU) to the network using different BNEP formats.

- Reference

  [5] 5.1, 5.2, 5.7, 6.1.1

  [1] 2.5, 2.9

- Initial Condition

  - Both the IUT (NAP) and the Lower Tester (PANU) are initialized.
  - Initialization may require a PIN if there are any security restrictions already in place on the Bluetooth link.
  - The IUT must have an open and configured BNEP connection with the Lower Tester.

- Test Procedure

  1. The Lower Tester transmits a BNEP packet to the IUT as described in Table 4.11.
  2. The IUT removes the BNEP header, adds Ethernet header, and multicasts the packet to the other network.
  3. The Upper Tester (Network Terminal) will receive the multicast packet.

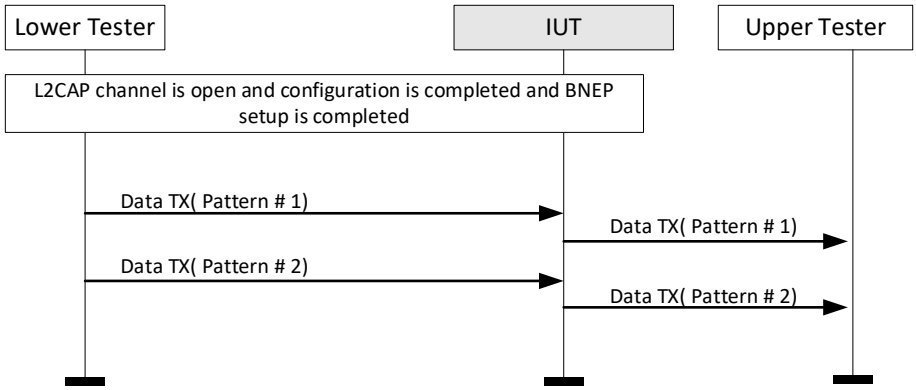

Figure 4.3: Multicasting from a Bluetooth Network to Other Networks

In step 1, the Lower Tester sends two test patterns as described below:

Pattern #1	**Value**
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	IEEE multicast address (01:00:5E:XX:XX:XX)
Source Address (48 bits)	Lower Tester (PANU)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Pattern #2	Value
L2CAP Length	0x5E5 (BNEP packet length within the L2CAP packet)
BNEP Type	0x04 (BNEP_COMPRESSED_ETHERNET_DEST_ONLY)
Destination Address (48 bits)	IEEE multicast address (01:00:5E:XX:XX:XX)
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.11: Multicasting from a Bluetooth Network to Other Networks – Data TX patterns

- Expected Outcome

  Pass verdict

  The Upper Tester receives two correct Ethernet multicast packets as described below (both test patterns from the Lower Tester to the IUT result in identical Ethernet packets on the Upper Tester):

Pattern #n	Value
Destination Address (48 bits)	IEEE multicast Multicast address (01:00:5E:XX:XX:XX)
Source Address (48 bits)	Lower Tester (PANU)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.12: Multicasting from a Bluetooth Network to Other Networks – Pass criteria

PAN/NAP/BNEP/MULTICAST-1/BV-04-C [Multicasting from Another Network to a Bluetooth Network]

- Test Purpose

  Verify the use of the BNEP_GENERAL_ETHERNET (0x00) packet when multicasting packets from the Upper Tester (Network Terminal) through an IUT (NAP) to the Lower Tester (PANU).

- Reference

  [5] 5.1, 5.2, 5.7, 6.1.1

  [1] 2.5

- Initial Condition

  - Both the IUT (NAP) and the Lower Tester (PANU) are initialized.
  - Initialization may require a PIN if there are any security restrictions already in place on the Bluetooth link.
  - The IUT has an open and configured BNEP connection with the Lower Tester. The Lower Tester is the acceptor in the Bluetooth connection, receiving BNEP packets from the IUT (multicast packets).

- Test Procedure

  1. The Upper Tester (Network Terminal) sends an Ethernet multicast packet; the test pattern is as described in Table 4.13.
  2. The IUT forwards the multicast Ethernet packet to the Lower Tester as a BNEP packet.

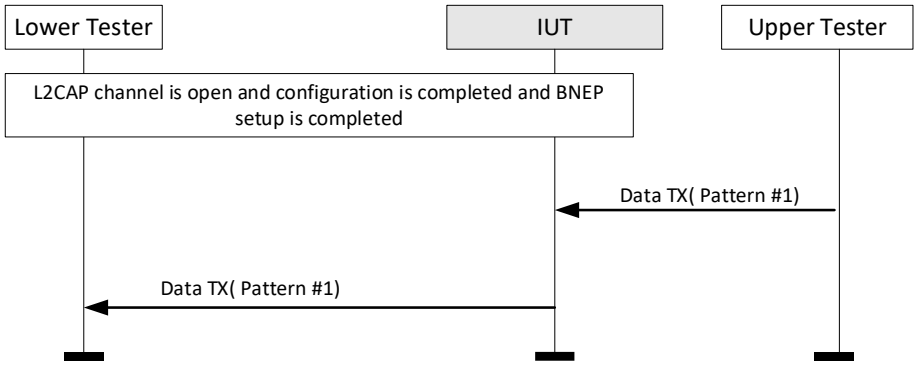

Figure 4.4: Multicasting from Another Network to a Bluetooth Network

In step 1, the Upper Tester sends a test pattern as described below:

Pattern #1	**Value**
Destination Address (48 bits)	IEEE multicast address (01:00:5E:XX:XX:XX)
Source Address (48 bits)	Upper Tester (Network Terminal's) Ethernet Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.13: Multicasting from Another Network to a Bluetooth Network – Data TX pattern

- Expected Outcome

  Pass verdict

  The IUT sends the packet described below to the Lower Tester:

Pattern #1	**Value**
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	Multicast address (01:00:5E:XX:XX:XX)
Source Address (48 bits)	Upper Tester (Network Terminal's) Ethernet Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.14: Multicasting from Another Network to a Bluetooth Network – Pass criteria

4.4 PAN Forwarding test cases

PAN/NAP/BNEP/FORWARD-UNICAST/BV-05-C [NAP Performing Layer 2 Bridging of a Unicast Packet from a Bluetooth PANU to Another Network Technology (e.g., Ethernet)]

- Test Purpose

  Verify that BNEP packets sent by the Lower Tester (PANU) to the Upper Tester (Network Terminal) are converted to the correct format for the non-Bluetooth network and delivered to the Upper Tester (Network Terminal).

- Reference

  [5] 5, 5.1, 5.2, 5.4

  [1] 2.5, 2.9

- Initial Condition

  - Network connectivity has been established between the Upper Tester (Network Terminal) and the IUT.
  - Both the IUT (NAP) and the Lower Tester (PANU) are initialized.
  - Initialization may require a PIN if there are any security restrictions already in place on the Bluetooth link.
  - Either the Lower Tester or the IUT has initiated the establishment and configuration of L2CAP and BNEP connections.
  - No filters are set on the BNEP connection.
  - The IUT has an open and configured BNEP connection with the Lower Tester.

- Test Procedure

  1. The Upper Tester (Network Terminal) listens for incoming packets and lists the packet it receives.
  2. The Lower Tester sends the IUT two unicast packets addressed to the Upper Tester (Network Terminal) at the appropriate layer 2 address as described in Table 4.15.
  3. To verify the integrity of the data, the Upper Tester lists the packets it receives during the course of the test.

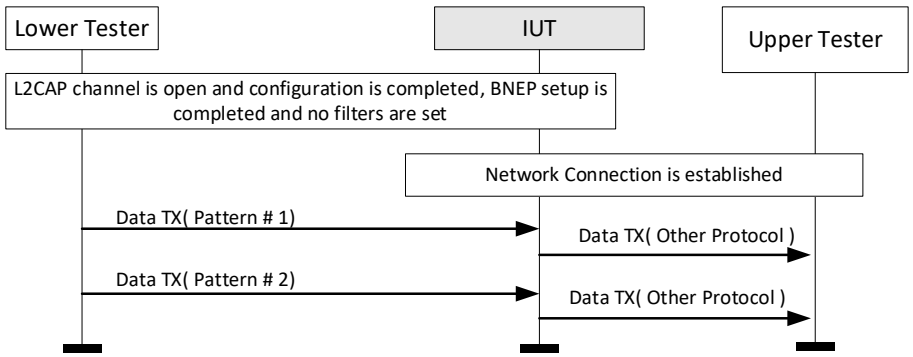

Figure 4.5: NAP Performing Layer 2 Bridging of a Unicast Packet from a Bluetooth PANU to Another Network Technology (e.g., Ethernet)

In step 2, the Lower Tester will send the following two packets as described below:

Pattern #1	**Value**
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	MAC address of the Upper Tester (Network Terminal)
Source Address	Lower Tester (PANU) Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Pattern #2	**Value**
L2CAP Length	0x5E5 (BNEP packet length within the L2CAP packet)
BNEP Type	0x04 (BNEP_COMPRESSED_ETHERNET_DEST_ONLY)
Destination Address (48 bits)	MAC address of the Upper Tester (Network Terminal)
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.15: NAP Performing Layer 2 Bridging of a Unicast Packet from a Bluetooth PANU to Another Network Technology (e.g., Ethernet) – Data TX patterns

- Expected Outcome

  Pass verdict

  Both of the packets sent by the Lower Tester are correctly converted by the IUT to the non-Bluetooth network format and forwarded to the Upper Tester.

PAN/NAP/BNEP/FORWARD-UNICAST/BV-06-C [NAP Performing Layer 2 Bridging of a Unicast Packet from a Non-Bluetooth Network to a Bluetooth PANU]

- Test Purpose

  Verify that data packets sent by the Upper Tester (Network Terminal) from the IUT (NAP) to the Lower Tester (PANU) are converted from the non-Bluetooth network packet format to valid BNEP packets and delivered to the Lower Tester.

- Reference

  [5] 5, 5.1, 5.2, 5.4

  [1] 2.5, 2.8

- Initial Condition

  - Network connectivity has been established between the Upper Tester (Network Terminal) and the IUT.
  - Both the IUT (NAP) and the Lower Tester (PANU) are initialized.

- Initialization may require a PIN if there are any security restrictions already in place on the Bluetooth link.
- Either the Lower Tester or the IUT has initiated the establishment and configuration of L2CAP and BNEP connections.
- No filters are set on the BNEP connection.
- The IUT has an open and configured BNEP connection with the Lower Tester.

- Test Procedure

1. The Lower Tester listens for incoming packets and lists the packet it receives.
2. The Upper Tester sends a unicast packet addressed to the Lower Tester's Bluetooth Address. The payload of the packet sent by the Upper Tester is the payload contained in the packet expected to be received by the Lower Tester, and it may be restricted in size equal to the MTU of the non-Bluetooth network.
3. To verify the integrity of the data in step 2, the Lower Tester lists the packets it receives during the course of the test.

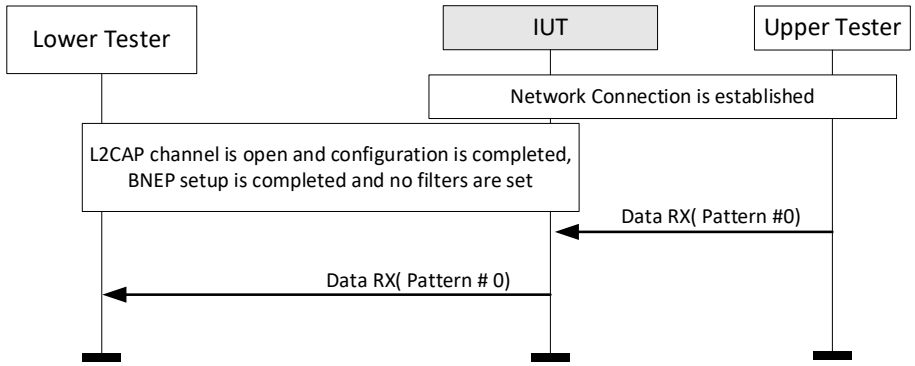

Figure 4.6: NAP Performing Layer 2 Bridging of a Unicast Packet from a Non-Bluetooth Network to a Bluetooth PANU

- Expected Outcome

Pass verdict

Data sent by the Upper Tester reaches the Lower Tester in a BNEP packet.

The packet may be truncated and the packet size adjusted accordingly, if the expected size exceeded the MTU of the other networking technology.

The Lower Tester receives either of the following packets:

Pattern #0 – ALT A	**Value**
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	Lower Tester (PANU) Bluetooth Address
Source Address (48 bits)	MAC address of the Upper Tester (Network Terminal)
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Or

Pattern #0 – ALT B	**Value**
L2CAP Length	0x5E5 (BNEP packet length within the L2CAP packet)
BNEP Type	0x04 (BNEP_COMPRESSED_ETHERNET_SOURCE_ONLY)
Source Address (48 bits)	MAC address of the Upper Tester (Network Terminal)
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.16: NAP Performing Layer 2 Bridging of a Unicast Packet from a Non- Bluetooth Network to a Bluetooth PANU – Pass criteria

4.4.1 Forward for BNEP Data Packets

- Test Purpose

  Verify the correct processing of the BNEP_GENERAL_ETHERNET (0x00) and BNEP_COMPRESSED_ETHERNET_DEST_ONLY (0x04) packet type for forwarding Ethernet packets in a Bluetooth piconet.

  The IUT receives the packet from the originator and forwards it to the destination node.

- Reference

  [5] 5, 5.1, 5.2, 5.4, 10.1

  [1] 2.5, 2.8, 2.9

- Initial Condition

  - This test involves three Bluetooth devices that form one piconet: IUT (NAP/GN) + two Lower Testers (PANU 1 and PANU 2). The IUT (NAP/GN) is the initiator in the Bluetooth connection. The two Lower Testers (PANU 1 and PANU 2) will act as clients.
  - The IUT (NAP/GN) and the two Lower Testers (PANU 1 and PANU 2) are initialized.
  - Initialization may require a PIN if there are any security restrictions already in place on the Bluetooth link.
  - The IUT has an open and configured BNEP connection with the Lower Testers. The IUT is the initiator and the Lower Testers (PANU 1 and PANU 2) are the acceptors in the Bluetooth connection.
  - There are no filters set with Lower Tester 1 (PANU 1) and Lower Tester 2 (PANU 2).
  - Lower Tester 1 (PANU 1) and Lower Tester 2 (PANU 2) do not have any direct connection with each other.
  - Lower Tester 1 (PANU 1) knows the BD_ADDR of Lower Tester 2 (PANU 2).

- Test Case Configuration

Test Case
PAN/NAP/BNEP/FORWARD/BV-08-C [Forward for BNEP Data Packets]
PAN/GN/BNEP/FORWARD/BV-08-C [Forward for BNEP Data Packets]

Table 4.17: Forward for BNEP Data Packets test cases

- Test Procedure

1. Lower Tester 1 (PANU 1) sends a packet containing pattern #1 and pattern #2 to the IUT.
2. The IUT receives each packet with pattern #1 and pattern #2 from Lower Tester 1 (PANU 1). The IUT processes each of the packets and forwards each packet to Lower Tester 2 (PANU 2).

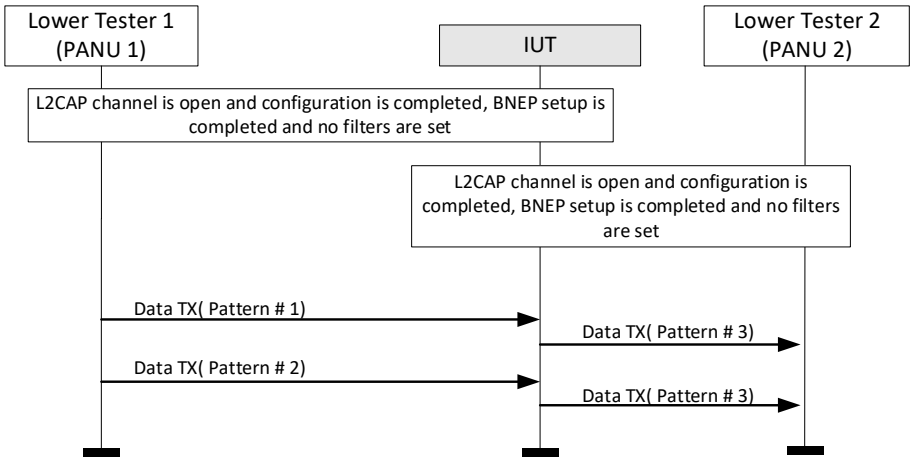

Figure 4.7: Forward for BNEP Data Packets

In step 1, Lower Tester 1 (PANU 1) sends a test pattern as described below:

Pattern #1	Value
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP Type field is BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	Lower Tester 2 (PANU 2)'s Bluetooth Address
Source Address (48 bits)	Lower Tester 1 (PANU 1)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Pattern #2	Value
L2CAP Length	0x5E5
BNEP Type	0x04 (BNEP_COMPRESSED_ETHERNET_DEST_ONLY)
Destination Address (48 bits)	Lower Tester 2 (PANU 2)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)

Pattern #2	Value
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.18: Forward for BNEP Data Packets – Data TX pattern Lower Tester 1

- Expected Outcome

  Pass verdict

  The IUT receives and interprets the BNEP header correctly sent from Lower Tester 1 (PANU 1) and subsequently forwards it to Lower Tester 2 (PANU 2).

  The packet sent to Lower Tester 2 (PANU 2) contains the data as one of the alternatives described below:

Pattern #3 – ALT A	Value
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	Lower Tester 2 (PANU 2)'s Bluetooth Address
Source Address (48 bits)	Lower Tester 1 (PANU 1)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

  Or

Pattern #3 – ALT B	Value
L2CAP Length	0x5E5 (BNEP packet length within the L2CAP packet)
BNEP Type	0x03 (BNEP_COMPRESSED_ETHERNET_SOURCE_ONLY)
Source Address (48 bits)	Lower Tester 1 (PANU 1)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.19: Forward for BNEP Data Packets – Pass criteria (interaction with PANU 2)

4.4.2 Forward for BNEP Broadcast Data Packets

- Test Purpose

  Verify the correct use of the BNEP_GENERAL_ETHERNET (0x00) and BNEP_COMPRESSED_ETHERNET_DEST_ONLY (0x04) packet type for forwarding Ethernet broadcast packets in a Bluetooth piconet.

  The IUT receives the packet from the originator and forwards it to its other PANUs, and the IUT also passes up a copy of the packet to its upper protocol layer.

- Reference

  [5] 5, 5.1, 5.2, 5.4, 5.7

  [1] 2.5, 2.9

- Initial Condition
  - This test involves four Bluetooth devices that form one piconet: IUT (NAP/GN) + three Lower Testers (PANU 1, PANU 2, and PANU 3). The IUT (NAP/GN) is the initiator in the Bluetooth connection. The three Lower Testers (PANU 1, PANU 2, and PANU 3) will act as clients.
  - The IUT (NAP/GN) and the three Lower Testers (PANU 1, PANU 2, and PANU 3) are initialized.
  - Initialization may require a PIN if there are any security restrictions already in place on the Bluetooth link.
  - The IUT has an open and configured BNEP connection with the Lower Testers. The IUT is the initiator and the Lower Testers (PANU 1, PANU 2, and PANU 3) are the acceptors in the Bluetooth connection.
  - There are no filters set with Lower Tester 1 (PANU 1), Lower Tester 2 (PANU 2), and Lower Tester 3 (PANU 3).
  - Lower Tester 1 (PANU 1), Lower Tester 2 (PANU 2), and Lower Tester 3 (PANU 3) do not have any direct connection with each other.
- Test Case Configuration

Test Case
PAN/NAP/BNEP/FORWARD-BROADCAST/BV-09-C [Forward for BNEP Broadcast Data Packets]
PAN/GN/BNEP/FORWARD-BROADCAST/BV-09-C [Forward for BNEP Broadcast Data Packets]

Table 4.20: Forward for BNEP Broadcast Data Packets test cases

- Test Procedure

  1. Lower Tester 1 (PANU 1) sends a packet containing pattern #1 and pattern #2 to the IUT.
  2. The IUT receives the packets with pattern #1 and pattern #2 from Lower Tester 1 (PANU 1).
  3. The IUT processes each of the packets and forwards each packet to Lower Tester 2 (PANU 2) and Lower Tester3 (PANU 3), as well as to its own upper protocol layers if applicable.

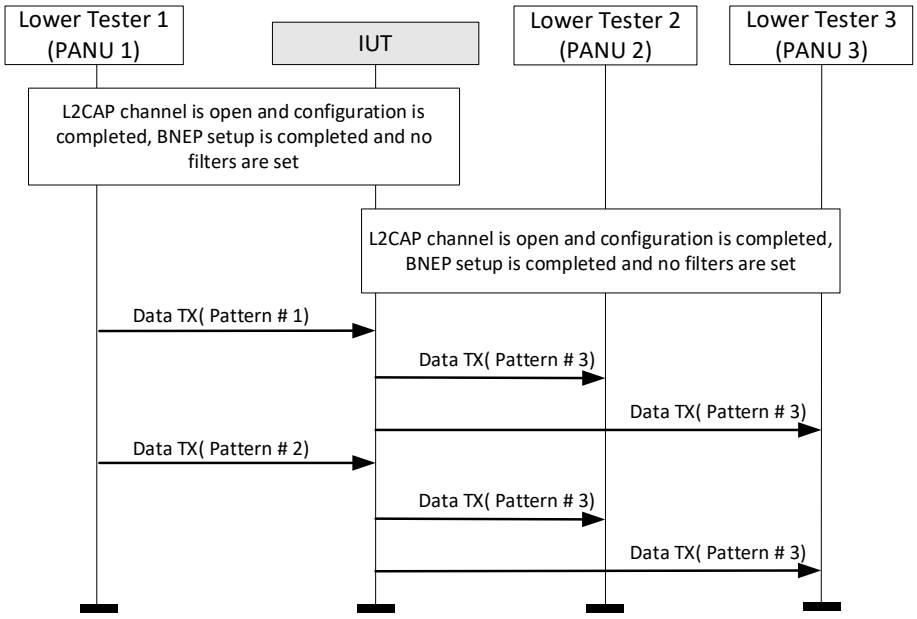

Figure 4.8: Forward for BNEP Broadcast Data Packets

In step 1, Lower Tester 1 (PANU 1) sends a test pattern as described below:

Pattern #1	**Value**
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address	FF:FF:FF:FF:FF:FF (Broadcast Address)
Source Address (48 bits)	Lower Tester 1 (PANU 1)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Pattern #2	**Value**
L2CAP Length	0x5E5 (BNEP packet length within the L2CAP packet)
BNEP Type	0x04 (BNEP_COMPRESSED_ETHERNET_DEST_ONLY)
Destination Address	FF:FF:FF:FF:FF:FF (Broadcast Address)
Network Protocol Type	0x0800, or 0x86DD, or 0xXXXX
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.21: Forward for BNEP Broadcast Data Packets – Data TX pattern Lower Tester 1

- Expected Outcome

  Pass verdict

  The IUT receives and interprets the BNEP header correctly and subsequently forwards it to Lower Tester 2 (PANU 2) and Lower Tester 3 (PANU 3) on the right BNEP connection.

  The IUT does not transmit a packet to Lower Tester 1 (PANU 1).

  The IUT correctly removes the BNEP packet info from the data and passes a valid packet to the upper protocol layer if the IUT supports the Network Protocol Type contained in the packet.

  The packet sent to Lower Tester 2 (PANU 2) and Lower Tester 3 (PANU 3) contains the data described below:

Pattern #3	**Value**
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP Type field is BNEP_GENERAL_ETHERNET)
Destination Address	FF:FF:FF:FF:FF:FF (Broadcast Address)
Source Address (48 bits)	Lower Tester 1 (PANU 1)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.22: Forward for BNEP Broadcast Data Packets – Pass criteria (interaction with PANU 2 and PANU 3)

4.5 PAN Extension Headers test cases

4.5.1 Sending an Ethernet Packet with Two Unknown Extension Headers

- Test Purpose

  Verify that the IUT (NAP/GN) responds to Lower Tester 1 (PANU 1), that the unknown control header is not known to the node, and that the IUT (NAP/GN) will forward the data payload and the unknown extension headers to the other Lower Tester (PANU 2) and the unknown control header will be dropped at the IUT (NAP/GN).

- Reference

  [5] 5, 5.1, 5.2, 5.4

  [1] 2.5, 2.8, 3.1

- Initial Condition

  - This test involves three Bluetooth devices (IUT (NAP/GN) + two Lower Testers (PANU 1 and PANU 2)). The IUT (NAP/GN) is the initiator in the Bluetooth connection. The two Lower Testers (PANU 1 and PANU 2) will act as clients.
  - The IUT (NAP/GN) and the two Lower Testers (PANU 1 and PANU 2) are initialized.
  - Initialization may require a PIN if there are any security restrictions already in place on the Bluetooth link.

- The IUT has an open and configured BNEP connection with the Lower Testers. The IUT is the initiator and the Lower Testers (PANU 1 and PANU 2) are the acceptors in the Bluetooth connection.

- Test Case Configuration

Test Case
PAN/NAP/BNEP/EXTENSION-0/BV-07-C [Sending an Ethernet Packet with Two Unknown Extension Headers]
PAN/GN/BNEP/EXTENSION-0/BV-07-C [Sending an Ethernet Packet with Two Unknown Extension Headers]

Table 4.23: Sending an Ethernet Packet with Two Unknown Extension Headers test cases

- Test Procedure

1. Lower Tester 1 (PANU 1) transmits a BNEP packet to the IUT.
2. The IUT removes the unknown control extension header and sends an indication for an unknown control message back to Lower Tester 1 (PANU 1).
3. The IUT forwards the data payload along with the two unknown extension headers to Lower Tester 2 (PANU 2).

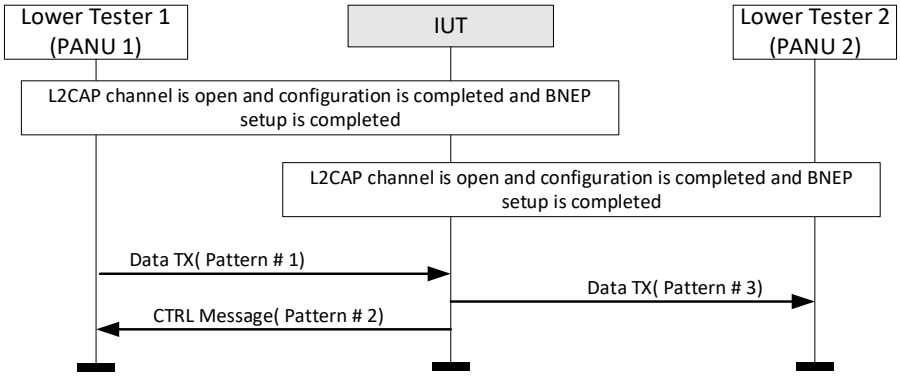

Figure 4.9: Sending an Ethernet Packet with Two Unknown Extension Headers

In step 1, Lower Tester 1 (PANU 1) sends a test pattern as described below:

Pattern #1	Value
L2CAP Length	0x069B (BNEP packet length within the L2CAP packet, equal to the BNEP minimum MTU = 1691 bytes)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
BNEP Type + Extension	0x80 BNEP_GENERAL_ETHERNET = 0x00 (7 bits) Extension = 0x1 (1 bit)
Destination Address (48 bits)	Lower Tester 2 (PANU 2)'s Bluetooth address
Source Address (48 bits)	Lower Tester 1 (PANU 1)'s Bluetooth address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Extension header + Extension	0x80 Extension header: 0x00 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF
Extension BNEP Control Type	0x55 (Unknown control type)
Unknown BNEP control payload	0xFE bytes

Pattern #1	Value
Extension header + Extension	0xD5 Extension header is 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01.0xFE)
Extension header + Extension	0x7F Extension header: 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF, 0xFE, .0x01)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x34D (845) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x4B, 0x4C}

Table 4.24: Sending an Ethernet Packet with Two Unknown Extension Headers – Data TX pattern Lower Tester 1

- Expected Outcome

  Pass verdict

  Lower Tester 1 (PANU 1) receives the correct CTRL message pattern #2 as described below:

Pattern #2	Value
L2CAP Length	0x0003 (BNEP packet length within the L2CAP packet)
BNEP Type + extension	0x01 BNEP Type = BNEP_CONTROL (0x01), extension = 0 (1 bit)
BNEP Control	0x00 (BNEP_CONTROL_COMMAND_NOT_UNDERSTOOD)
Unknown control type	0x55 (BNEP control type that was received and not understood)

Table 4.25: Sending an Ethernet Packet with Two Unknown Extension Headers – Pass criteria (interaction with PANU 1)

Lower Tester 2 (PANU 2) receives the correct data packet with extension headers with either pattern #3 as described below:

Pattern #3 – ALT A	Value
L2CAP Length	0x059A (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x80 BNEP_GENERAL_ETHERNET = 0x00 (7 bits) Extension = 0x1 (1 bit)
Destination Address (48 bits)	Lower Tester 2 (PANU 2)’s Bluetooth Address
Source Address (48 bits)	Lower Tester 1 (PANU 1)’s Bluetooth address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Extension header + Extension	0xD5 Extension header: 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF

Pattern #3 – ALT A	**Value**
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00, 0x01..0xFE)
Extension header + Extension	0x7F Extension header is 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF, 0xFE..0x01)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x34D (845) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x4B, 0x4C}

Or

Pattern #3 – ALT B	**Value**
L2CAP Length	0x0594 (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x83 BNEP Type: BNEP_COMPRESSED_ETHERNET_SOURCE_ONLY = 0x03 (7 bits), Extension = 0x1 (1 bit)
Source Address (48 bits)	Lower Tester 1 (PANU 1)'s Bluetooth address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Extension header + Extension	0xD5 Extension header is 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00, 0x01..0xFE)
Extension header + Extension	0x7F Extension header: 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF, 0xFE..0x01)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x34D (845) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x4B, 0x4C}

Table 4.26: Sending an Ethernet Packet with Two Unknown Extension Headers – Pass criteria (interaction with PANU 2)

4.6 PAN Filtering test cases

4.6.1 Set Multicast Address Filter

- Test Purpose

  Verify that the IUT (NAP/GN) correctly handles receiving BNEP_FILTER_MULTI_ADDR_SET_MSG from Lower Tester 2 (PANU).

  Lower Tester 1 sets a Multicast Address Filter in the IUT, and sends several packets to the IUT. The IUT receives the packets from the originator and forwards those packets to the destination node, if they have not been filtered out.

- Reference

  [5] 5.1, 5.2, 5.4, 5.7

  [1] 2.6.1, 2.6.6

- Initial Condition

  - This test involves three Bluetooth devices (IUT (NAP/GN) + two Lower Testers (PANU 1 and PANU 2)). The IUT (NAP/GN) is the initiator in the Bluetooth connection. The two Lower Testers (PANU 1 and PANU 2) will act as clients.
  - The IUT (NAP/GN) and the two Lower Testers (PANU 1 and PANU 2) are initialized.
  - Initialization may require entering a PIN if there are any security restrictions already in place on the Bluetooth link.
  - The IUT has an open and configured BNEP channel with the Lower Testers. Lower Tester 1 (PANU 1) and Lower Tester 2 (PANU 2) do not have any direct connection with each other. Lower Tester 1 (PANU 1) knows about the BD_ADDR of Lower Tester 2 (PANU 2).
  - For this test, a base multicast address must be chosen and must be a valid multicast address (for example, 0x030002300000); all test patterns use this base address plus an integer offset to determine which multicast address to use in each test pattern. For example, if a multicast address of 0x030002300000 is chosen as the base address, then if a test pattern states a value of Base Address + 2, then the value for the multicast address for the test pattern will be 0x030002300002.

- Test Case Configuration

Test Case
PAN/NAP/BNEP/FILTER/BV-10-C [Set Multicast Address Filter]
PAN/GN/BNEP/FILTER/BV-10-C [Set Multicast Address Filter]

Table 4.27: Set Multicast Address Filter test cases

- Test Procedure

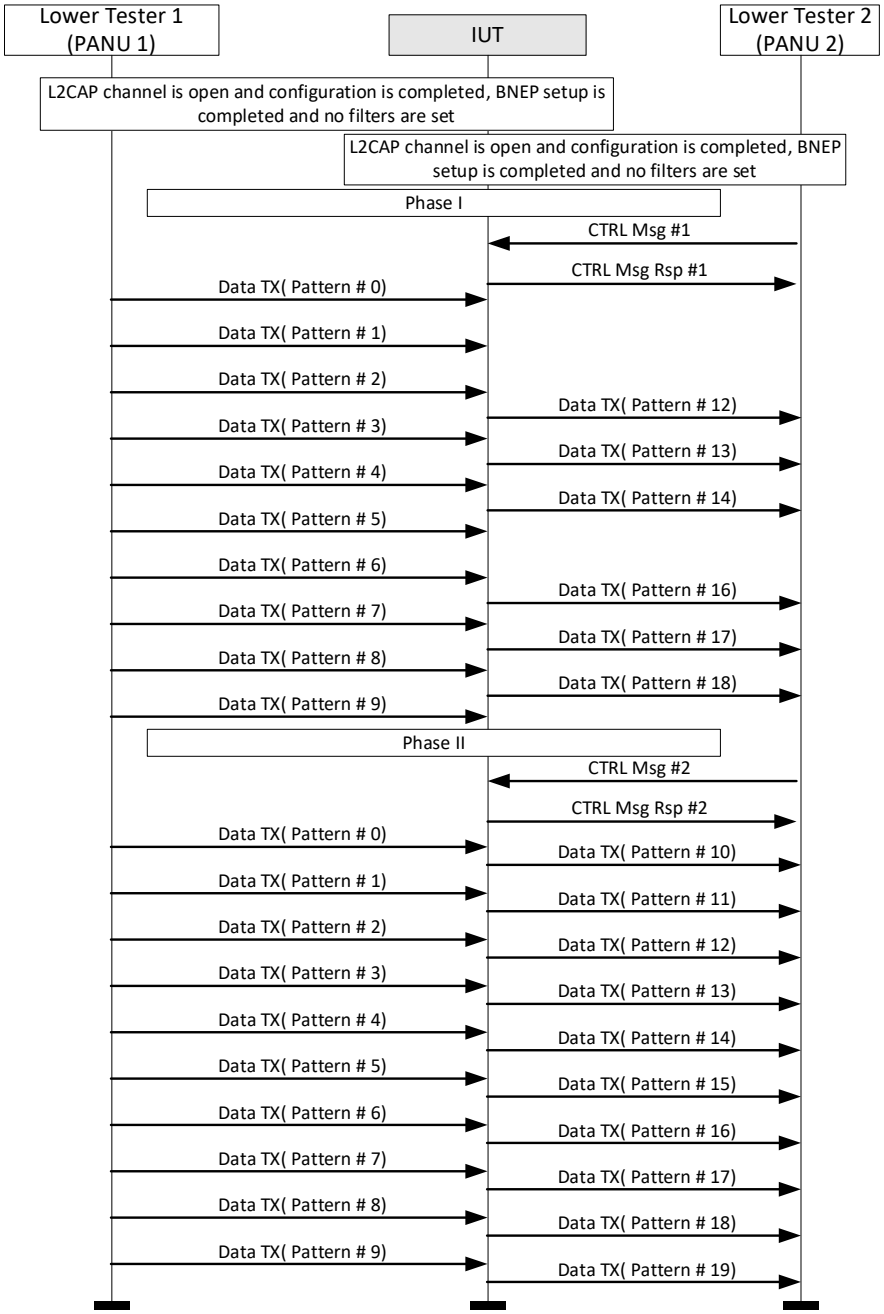

Figure 4.10: Set Multicast Address Filter

In Phase I, the IUT (NAP/GN) receives a BNEP_FILTER_MULTI_ADDR_SET_MSG containing two filter ranges from Lower Tester 2 (PANU) and responds to this command by sending a BNEP_FILTER_MULTI_ADDR_RESPONSE_MSG back to Lower Tester 2 (PANU). If the response is positive, Lower Tester 1 (PANU) starts sending the test pattern #0 – #9 to the IUT (NAP/GN), which has to forward to Lower Tester 2 (PANU) based on the filter settings.

In Phase II, the IUT (NAP/GN) receives a BNEP_FILTER_MULTI_ADDR_SET_MSG, which resets the filter ranges from Lower Tester 2 (PANU) and responds to this command by sending a BNEP_FILTER_MULTI_ADDR_RESPONSE_MSG back to Lower Tester 2 (PANU). If the response is

positive, Lower Tester 1 (PANU) starts sending the test pattern #0 – #9 to the IUT (NAP/GN), which has to forward to Lower Tester 2 (PANU) based on the reset filter settings.

Phase I

Lower Tester 2 (PANU) sends the command below:

CTRL Msg #1	**Value**
L2CAP Length	0x001C (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (BNEP Extension flag = 0x0 = no extension header)
BNEP Control Type	0x05 (BNEP_FILTER_MULTI_ADDR_SET_MSG)
List Length	0x0018 Length of the list, in bytes, of the multicast address not to be filtered
Multicast Address Range Start #1	Base Address + 2
Multicast Address Range End #1	Base Address + 4
Multicast Address Range Start #2	Base Address + 6
Multicast Address Range End #2	Base Address + 8

Table 4.28: Set Multicast Address Filter, Phase I, Lower Tester 2 command

Lower Tester 1 (PANU) sends the patterns below.

Pattern #0–9	**Value**
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP Type field is BNEP_GENERAL_ETHERNET)
Destination Address	Multicast address depending on which test pattern: Pattern 0: Base Address + 0 Pattern 1: Base Address + 1 Pattern 2: Base Address + 2 Pattern 3: Base Address + 3 Pattern 4: Base Address + 4 Pattern 5: Base Address + 5 Pattern 6: Base Address + 6 Pattern 7: Base Address + 7 Pattern 8: Base Address + 8 Pattern 9: Base Address + 9
Source Address (48 bits)	Lower Tester 1 (PANU)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.29: Set Multicast Address Filter, Phase I, Lower Tester 1 patterns

Phase II

Lower Tester 2 (PANU) sends the command below:

CTRL Msg #2	Value
L2CAP Length	0x0004 (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (No extension header)
BNEP Control Type	0x05 (BNEP_FILTER_MULTI_ADDR_SET_MSG)
List Length	0x0000 (Reset filters)

Table 4.30: Set Multicast Address Filter, Phase II, Lower Tester 2 command

Lower Tester 1 (PANU) sends the patterns below:

Pattern #0–9	Value
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address	Multicast address depending on which test pattern: Pattern 0: Base Address + 0 Pattern 1: Base Address + 1 Pattern 2: Base Address + 2 Pattern 3: Base Address + 3 Pattern 4: Base Address + 4 Pattern 5: Base Address + 5 Pattern 6: Base Address + 6 Pattern 7: Base Address + 7 Pattern 8: Base Address + 8 Pattern 9: Base Address + 9
Source Address (48 bits)	Lower Tester 1 (PANU)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.31: Set Multicast Address Filter, Phase II, Lower Tester 1 patterns

- Expected Outcome

  Pass verdict

  Phase I

  The IUT (NAP/GN) receives and interprets the BNEP_FILTER_MULTI_ADDR_SET_MSG from Lower Tester 2 (PANU) and responds to that correctly with the following message:

CTRL Msg RSP #1	Value	Description
L2CAP Length	0x0004	

CTRL Msg RSP #1	Value	Description
BNEP Type	0x01	BNEP Type field is BNEP_CONTROL
Extension	0x0	BNEP Extension flag = 0x0 = no extension header
BNEP Control Type	0x06	BNEP_FILTER_MULTI_ADDR_RESPONSE_MSG
Response	0x0000	Operation Successful

Table 4.32: Set Multicast Address Filter, Phase I, Pass criteria (message)

The IUT (NAP/GN) forwards only test patterns 12, 13, 14, 16, 17, and 18 to Lower Tester 2 (PANU). The packages sent to Lower Tester 2 (PANU) contain the data below:

Pattern #n	Value
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address	Multicast address depending on which test pattern: Pattern 12: Base Address + 2 Pattern 13: Base Address + 3 Pattern 14: Base Address + 4 Pattern 16: Base Address + 6 Pattern 17: Base Address + 7 Pattern 18: Base Address + 8
Source Address (48 bits)	Lower Tester 1 (PANU)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.33: Set Multicast Address Filter, Phase I, Pass criteria (data)

Phase II

The IUT (NAP/GN) receives and interprets the BNEP_FILTER_MULTI_ADDR_SET_MSG from Lower Tester 2 (PANU) and responds to that correctly with the following message:

CTRL Msg RSP #1	Value
L2CAP Length	0x0004 (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (No extension header)
BNEP Control Type	0x06 (BNEP_FILTER_MULTI_ADDR_RESPONSE_MSG)
Response	0x0000 (Operation Successful)

Table 4.34: Set Multicast Address Filter, Phase II, Pass criteria (message)

The IUT (NAP/GN) forwards all of the test patterns #10 – #19 to Lower Tester 2 (PANU). The packages sent to Lower Tester 2 (PANU) contain the data below:

Pattern #n	**Value**
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address	Multicast address depending on which test pattern: Pattern 10: Base Address + 0 Pattern 11: Base Address + 1 Pattern 12: Base Address + 2 Pattern 13: Base Address + 3 Pattern 14: Base Address + 4 Pattern 15: Base Address + 5 Pattern 16: Base Address + 6 Pattern 17: Base Address + 7 Pattern 18: Base Address + 8 Pattern 19: Base Address + 9
Source Address (48 bits)	Lower Tester 1 (PANU)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.35: Set Multicast Address Filter, Phase II, Pass criteria (data)

4.6.2 Set Network Protocol Filter

- Test Purpose

  Verify that the IUT (NAP/GN) correctly handles the BNEP_FILTER_NET_TYPE_SET_MSG.

  One of the Lower Tester PANUs, Lower Tester 1, will set a Network Protocol Address Filter in the IUT (NAP/GN). The other Lower Tester PANU, Lower Tester 2, will send several packets to the IUT (NAP/GN). The IUT (NAP/GN) receives the packets from the originator and forwards those packets to the destination node, if they have not been filtered out.

- Reference

  [5] 5.1, 5.2, 5.4

  [1] 2.6.1, 2.6.3, 2.6.5

- Initial Condition

  - This test involves three Bluetooth devices (IUT (NAP/GN) + two Lower Testers (PANU 1 and PANU 2)). The IUT (NAP/GN) is the initiator in the Bluetooth connection. The two Lower Testers (PANU 1 and PANU 2) will act as clients.
  - The IUT (NAP/GN) and the two Lower Testers (PANU 1 and PANU 2) are initialized.
  - Initialization may require entering a PIN if there are any security restrictions already in place on the Bluetooth link.

- The IUT has an open and configured BNEP channel with the Lower Testers. Lower Tester 1 (PANU 1) and Lower Tester 2 (PANU 2) do not have any direct connection with each other. Lower Tester 1 (PANU 1) knows about the BD_ADDR of Lower Tester 2 (PANU 2).
- For this test, a base networking protocol type must be chosen and must be a networking protocol type (for example, 0x0800); all test patterns use this base networking protocol type plus an integer offset to determine which networking protocol type to use in each test pattern. For example, if a networking protocol type of 0x0800 is chosen as the base networking protocol type, then if a test pattern states a value of base Networking Protocol Type + 2, then the value for the networking protocol type for the test pattern will be 0x0802.

- Test Case Configuration

Test Case
PAN/NAP/BNEP/FILTER/BV-11-C [Set Network Protocol Filter]
PAN/GN/BNEP/FILTER/BV-11-C [Set Network Protocol Filter]

Table 4.36: Set Network Protocol Filter test cases

- Test Procedure

  In Phase I, the IUT (NAP/GN) receives a BNEP_FILTER_NET_TYPE_SET_MSG containing two filter ranges from Lower Tester 2 (PANU 2) and responds to this command by sending a BNEP_FILTER_NET_TYPE_RESPONSE_MSG back to Lower Tester 2 (PANU 2). If the response is positive, Lower Tester 1 (PANU 1) starts sending the test pattern #0 – #9 to the IUT (NAP/GN), which has to forward to Lower Tester 2 (PANU 2) based on the filter settings.

  In Phase II, the IUT (NAP/GN) receives a BNEP_FILTER_NET_TYPE_SET_MSG, which resets the filter ranges from Lower Tester 2 (PANU 2) and responds to this command by sending a BNEP_FILTER_NET_TYPE_RESPONSE_MSG back to Lower Tester 2 (PANU 2). If the response is positive, Lower Tester 1 (PANU 1) starts sending the test pattern #0 – #9 to the IUT (NAP/GN), which has to forward to Lower Tester 2 (PANU 2) based on the reset filter settings.

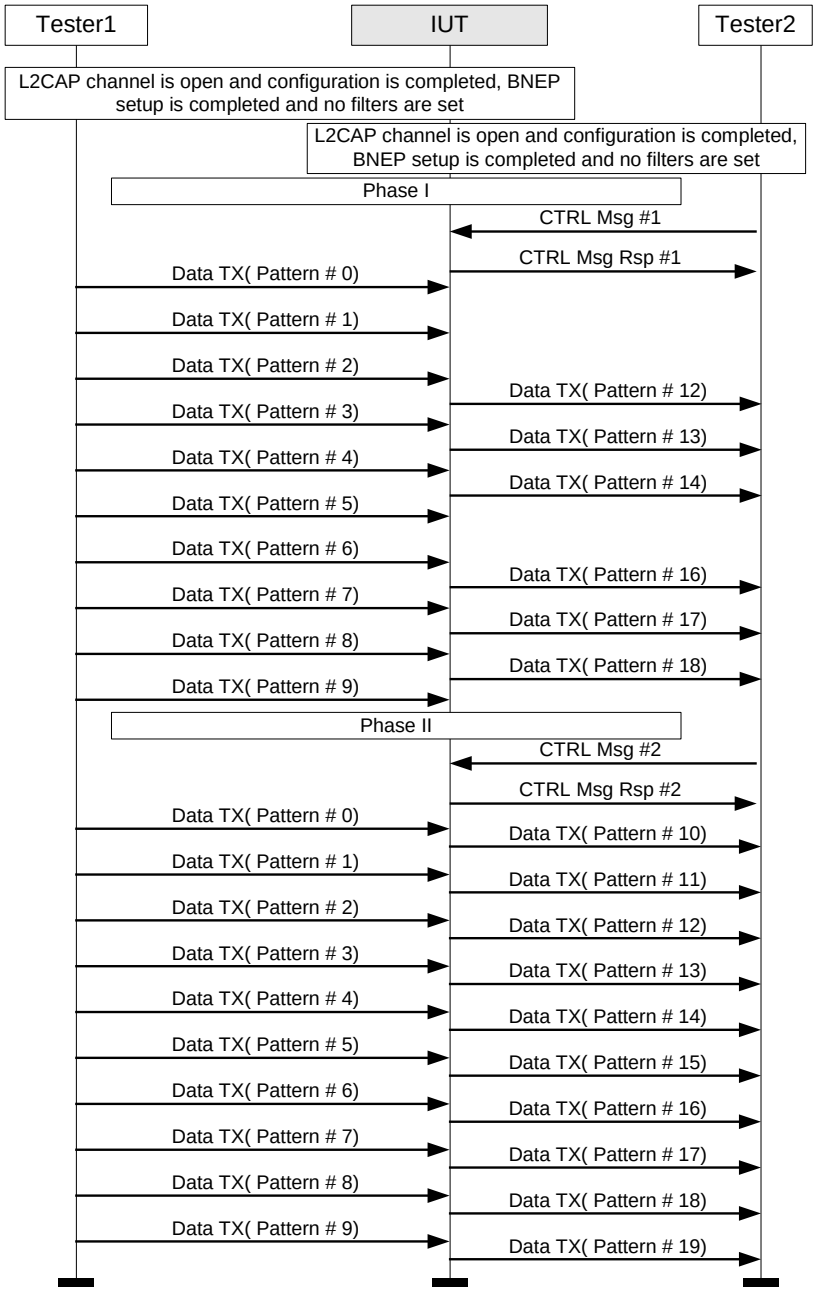

Figure 4.11: Set Network Protocol Filter

Phase I

Lower Tester 2 (PANU 2) sends the command below:

CTRL Msg #1	**Value**
L2CAP Length	0x000C (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (BNEP Extension flag = no extension header)
BNEP Control Type	0x03 (BNEP_FILTER_NET_TYPE_SET_MSG)
List Length	0x0008 (Length of the list, in bytes, of the network protocols not to be filtered)

CTRL Msg #1	Value
Network Protocol Type Range Start #1	Base Networking Protocol Type + 2
Network Protocol Type Range End #1	Base Networking Protocol Type + 4
Network Protocol Type Range Start #2	Base Networking Protocol Type + 6
Network Protocol Type Range End #2	Base Networking Protocol Type + 8

Table 4.37: Set Network Protocol Filter, Phase I, Lower Tester 2 command

Lower Tester 1 (PANU 1) sends the test patterns below:

Pattern #0–9	Value
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	Lower Tester 2 (PANU 2)'s Bluetooth Address
Source Address (48 bits)	Lower Tester 1 (PANU 1)'s Bluetooth Address
Network Protocol Type	Networking Protocol Type depending on which test pattern: Pattern 0: Base Networking Protocol Type + 0 Pattern 1: Base Networking Protocol Type + 1 Pattern 2: Base Networking Protocol Type + 2 Pattern 3: Base Networking Protocol Type + 3 Pattern 4: Base Networking Protocol Type + 4 Pattern 5: Base Networking Protocol Type + 5 Pattern 6: Base Networking Protocol Type + 6 Pattern 7: Base Networking Protocol Type + 7 Pattern 8: Base Networking Protocol Type + 8 Pattern 9: Base Networking Protocol Type + 9
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.38: Set Network Protocol Filter, Phase I, Lower Tester 1 test patterns

Phase II

Lower Tester 2 (PANU 2) sends the command below:

CTRL Msg #2	Value
L2CAP Length	0x0004
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (No extension header)
BNEP Control Type	0x03 (BNEP_FILTER_NET_TYPE_SET_MSG)
List Length	0x0000 (Reset filters)

Table 4.39: Set Network Protocol Filter, Phase II, Lower Tester 2 command

Lower Tester 1 (PANU 1) sends the test patterns below:

Pattern #0–9	**Value**
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	Lower Tester 2 (PANU 2)'s Bluetooth Address
Source Address (48 bits)	Lower Tester 1 (PANU 1)'s Bluetooth Address
Network Protocol Type	Networking Protocol Type depending on which test pattern: Pattern 0: Base Networking Protocol Type + 0 Pattern 1: Base Networking Protocol Type + 1 Pattern 2: Base Networking Protocol Type + 2 Pattern 3: Base Networking Protocol Type + 3 Pattern 4: Base Networking Protocol Type + 4 Pattern 5: Base Networking Protocol Type + 5 Pattern 6: Base Networking Protocol Type + 6 Pattern 7: Base Networking Protocol Type + 7 Pattern 8: Base Networking Protocol Type + 8 Pattern 9: Base Networking Protocol Type + 9
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.40: Set Network Protocol Filter, Phase II, Lower Tester 1 test patterns

- Expected Outcome

  Pass verdict

  Phase I

  The IUT (NAP/GN) receives and interprets the BNEP_FILTER_NET_TYPE_SET_MSG from Lower Tester 2 (PANU 2) and responds to that correctly with the following message:

CTRL Msg RSP #1	**Value**
L2CAP Length	0x0004 (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (BNEP Extension flag = no extension header)
BNEP Control Type	0x04 (BNEP_FILTER_NET_TYPE_RESPONSE_MSG)
Response	0x0000 (Operation Successful)

Table 4.41: Set Network Protocol Filter, Phase I, Pass criteria (message)

The IUT (NAP/GN) forwards only test pattern 12, 13, 14, 16, 17, and 18 to Lower Tester 2 (PANU). The packages sent to Lower Tester 2 (PANU 2) contains the data below:

Pattern #n	**Value**
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	Lower Tester 2 (PANU 2)'s Bluetooth Address
Source Address (48 bits)	Lower Tester 1 (PANU 1)'s Bluetooth Address

Pattern #n	Value
Network Protocol Type	Networking Protocol Type depending on which test pattern: Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.42: Set Network Protocol Filter, Phase I, Pass criteria (data, option 1)

Or

Pattern #n	Value
L2CAP Length	0x5E5 (BNEP packet length within the L2CAP packet)
BNEP Type	0x03 (BNEP_COMPRESSED_ETHERNET_SOURCE_ONLY)
Source Address (48 bits)	Lower Tester 1 (PANU 1)'s Bluetooth Address
Network Protocol Type	Networking Protocol Type depending on which test pattern: Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.43: Set Network Protocol Filter, Phase I, Pass criteria (data, option 2)

Phase II

The IUT (NAP/GN) receives and interprets the BNEP_FILTER_NET_TYPE_SET_MSG from Lower Tester 2 (PANU 2) and responds to that correctly with the following message:

CTRL Msg RSP #1	Value
L2CAP Length	0x0004 (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (BNEP Extension flag = no extension header)
BNEP Control Type	0x04 (BNEP_FILTER_NET_TYPE_RESPONSE_MSG)
Response	0x0000 (Operation Successful)

Table 4.44: Set Network Protocol Filter, Phase II, Pass criteria (message)

The IUT (NAP/GN) forwards all of the test patterns #10 – #19 to Lower Tester 2 (PANU 2). The packages sent to Lower Tester 2 (PANU 2) contain the data below:

Pattern #n	**Value**
L2CAP Length	0x5EB (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	Lower Tester 2 (PANU 2)'s Bluetooth Address
Source Address (48 bits)	Lower Tester 1 (PANU 1)'s Bluetooth Address
Network Protocol Type	Networking Protocol Type depending on which test pattern: Pattern 10: Base Networking Protocol Type + 0 Pattern 11: Base Networking Protocol Type + 1 Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 15: Base Networking Protocol Type + 5 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8 Pattern 19: Base Networking Protocol Type + 9
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.45: Set Network Protocol Filter, Phase II, Pass criteria (data, option 1)

Or

Pattern #n	**Value**
L2CAP Length	0x5E5 (BNEP packet length within the L2CAP packet)
BNEP Type	0x03 (BNEP_COMPRESSED_ETHERNET_SOURCE_ONLY)
Source Address (48 bits)	Lower Tester 1 (PANU 1)'s Bluetooth Address
Network Protocol Type	Networking Protocol Type depending on which test pattern: Pattern 10: Base Networking Protocol Type + 0 Pattern 11: Base Networking Protocol Type + 1 Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 15: Base Networking Protocol Type + 5 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8 Pattern 19: Base Networking Protocol Type + 9
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.46: Set Network Protocol Filter, Phase II, Pass criteria (data, option 2)

4.6.3 Network Protocol Filtering with 802.1Q Tag Header Packets

- Test Purpose

  Verify the correct implementation of network protocol filtering with 802.1Q tag header packets. The IUT (NAP/GN), which is piconet central, is the target device of this test.

  This test involves three Bluetooth devices that form one piconet. Thus, two testers (PANUs) (peripherals) are connected to the IUT (NAP/GN) that is the piconet central. One of the PANUs, Tester1, will set a Network Protocol Address Filter in the IUT (NAP/GN). The other PANU will send several packets to the IUT (NAP/GN). The IUT (NAP/GN) receives the packets from the originator and forwards those packets to the destination node, if they have not been filtered out.

- Reference

  BNEP [1] and L2CAP in [2]

- Initial Condition

  - Tester1 (PANU), Tester2 (PANU), and the IUT (NAP/GN) must be switched on and initialized.
  - Initialization may require entering a PIN if there are any security restrictions already in place on the Bluetooth link.
  - Tester1 (PANU), Tester2 (PANU), and the IUT (NAP/GN) must have valid Bluetooth device addresses (BD_ADDR).
  - The IUT (NAP/GN) must have an open and configured BNEP channel with Tester1 (PANU) and with Tester2 (PANU).
  - Tester1 (PANU) and Tester2 (PANU) must not have any direct connection with each other. Tester1 (PANU) must know about the BD_ADDR of Tester2 (PANU).
  - For this test, a base networking protocol type must be chosen and must be a networking protocol type (for example, 0x0800); all test patterns use this base networking protocol type plus an integer offset to determine which networking protocol type to use in each test pattern. For example, if a networking protocol type of 0x0800 is chosen as the base networking protocol type, then if a test pattern states a value of base Networking Protocol Type + 2, then the value for the networking protocol type for the test pattern will be 0x0802.

- Test Case Configuration

Test Case
PAN/NAP/BNEP/FILTER/BV-12-C [Network Protocol Filtering with 802.1Q Tag Header Packets]
PAN/GN/BNEP/FILTER/BV-12-C [Network Protocol Filtering with 802.1Q Tag Header Packets]

Table 4.47: Network Protocol Filtering with 802.1Q Tag Header Packets test cases

- Test Procedure

  In Phase I, the IUT (NAP/GN) receives a BNEP_FILTER_NET_TYPE_SET_MSG containing two filter ranges from Tester2 (PANU) and responds to this command by sending a BNEP_FILTER_NET_TYPE_RESPONSE_MSG back to Tester2 (PANU). If the response is positive, Tester1 (PANU) starts sending the test pattern #0 – #9 to the IUT (NAP/GN), which has to forward to Tester2 (PANU) based on the filter settings.

  In Phase II, the IUT (NAP/GN) receives a BNEP_FILTER_NET_TYPE_SET_MSG, which resets the filter ranges from Tester2 (PANU) and responds to this command by sending a BNEP_FILTER_NET_TYPE_RESPONSE_MSG back to Tester2 (PANU). If the response is positive,

Tester1 (PANU) starts sending the test pattern #0 – #9 to the IUT (NAP/GN), which has to forward to Tester2 (PANU) based on the reset filter settings.

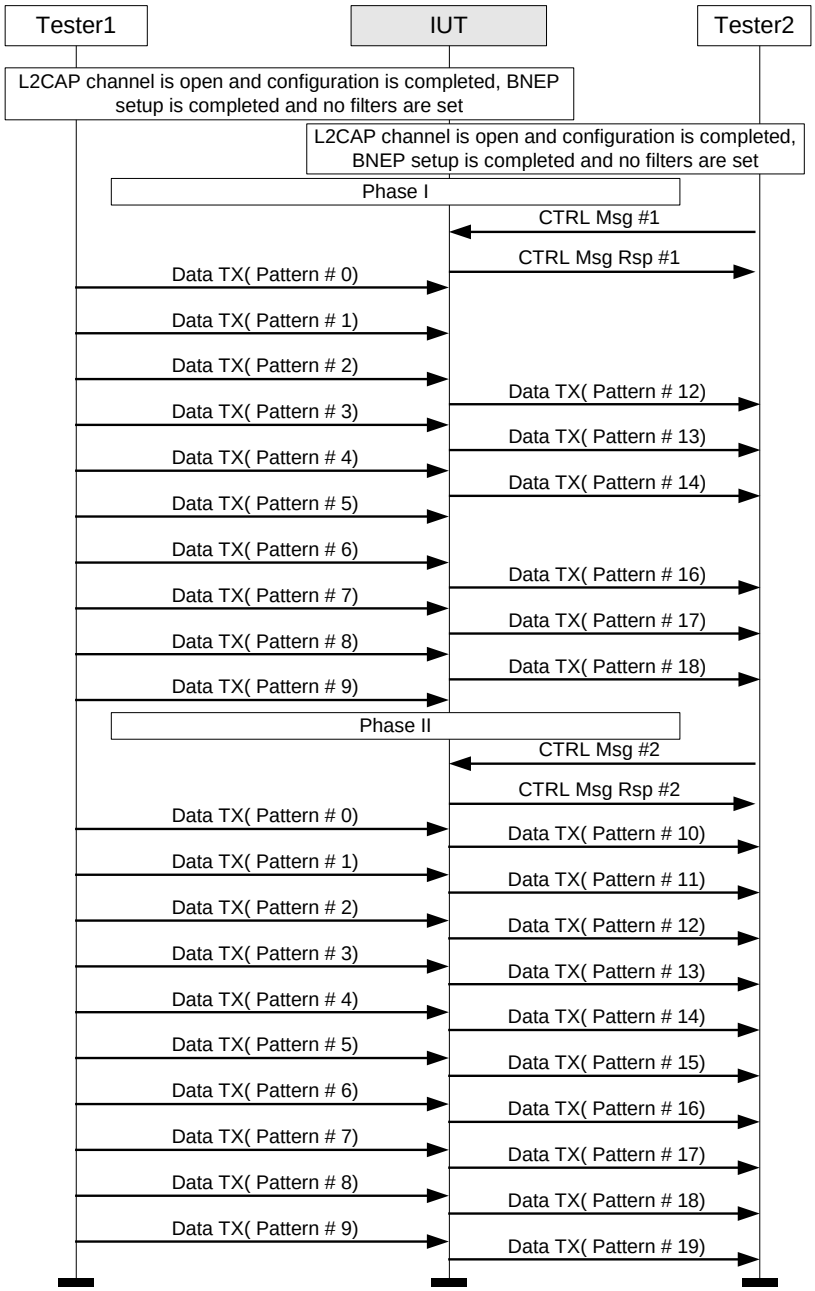

Figure 4.12: Network Protocol Filtering with 802.1Q Tag Header Packets

Phase I

Tester2 (PANU) sends the command below:

CTRL Msg #1	**Value**
L2CAP Length	0x000C (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (BNEP Extension flag = no extension header)
BNEP Control Type	0x03 (BNEP_FILTER_NET_TYPE_SET_MSG)

CTRL Msg #1	Value
List Length	0x0008 (Length of the list, in bytes, of the network protocols not to be filtered)
Network Protocol Type Range Start #1	Base Networking Protocol Type + 2
Network Protocol Type Range End #1	Base Networking Protocol Type + 4
Network Protocol Type Range Start #2	Base Networking Protocol Type + 6
Network Protocol Type Range End #2	Base Networking Protocol Type + 8

Table 4.48: Network Protocol Filtering with 802.1Q Tag Header Packets, Phase I, Tester2 command

Tester1 (PANU) sends the test patterns below:

Pattern #0–9	Value
L2CAP Length	0x5EF (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	Tester2 (PANU)'s Bluetooth Address
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x8100 (IEEE 802.1Q tag header)
Data [0-1]	0xXXXX IEEE 802.1Q tag header (Device MAY interpret the 802.1Q tag header)
Data [2-3]	Networking Protocol Type depending on which test pattern: Pattern 0: Base Networking Protocol Type + 0 Pattern 1: Base Networking Protocol Type + 1 Pattern 2: Base Networking Protocol Type + 2 Pattern 3: Base Networking Protocol Type + 3 Pattern 4: Base Networking Protocol Type + 4 Pattern 5: Base Networking Protocol Type + 5 Pattern 6: Base Networking Protocol Type + 6 Pattern 7: Base Networking Protocol Type + 7 Pattern 8: Base Networking Protocol Type + 8 Pattern 9: Base Networking Protocol Type + 9
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.49: Network Protocol Filtering with 802.1Q Tag Header Packets, Phase I, Tester1 test patterns

Phase II

Tester2 (PANU) sends the command below:

CTRL Msg #2	Value
L2CAP Length	0x0004 (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)

CTRL Msg #2	Value
Extension	0x0 (BNEP Extension flag = no extension header)
BNEP Control Type	0x03 (BNEP_FILTER_NET_TYPE_SET_MSG)
List Length	0x0000 (Length = 0 = Reset filters)

Table 4.50: Network Protocol Filtering with 802.1Q Tag Header Packets, Phase II, Tester2 command

Tester1 (PANU) sends the test patterns below:

Pattern #0–9	Value
L2CAP Length	0x5EF (BNEP packet length within the L2CAP packet)
BNEP Type	0x00
Destination Address (48 bits)	Tester2 (PANU)'s Bluetooth Address
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x8100 (IEEE 802.1Q tag header)
Data [0-1]	0xXXXX (Device MAY interpret the 802.1Q tag header)
Data [2-3]	Networking Protocol Type depending on which test pattern: Pattern 0: Base Networking Protocol Type + 0 Pattern 1: Base Networking Protocol Type + 1 Pattern 2: Base Networking Protocol Type + 2 Pattern 3: Base Networking Protocol Type + 3 Pattern 4: Base Networking Protocol Type + 4 Pattern 5: Base Networking Protocol Type + 5 Pattern 6: Base Networking Protocol Type + 6 Pattern 7: Base Networking Protocol Type + 7 Pattern 8: Base Networking Protocol Type + 8 Pattern 9: Base Networking Protocol Type + 9
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.51: Network Protocol Filtering with 802.1Q Tag Header Packets, Phase II, Tester1 test patterns

- Expected Outcome

  Pass verdict

  Phase I

  The IUT (NAP/GN) receives and interprets the BNEP_FILTER_NET_TYPE_SET_MSG from Tester2 (PANU) and responds to that correctly with the following message:

CTRL Msg RSP #1	Value
L2CAP Length	0x0004 (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (No extension header)
BNEP Control Type	0x04 (BNEP_FILTER_NET_TYPE_RESPONSE_MSG)

CTRL Msg RSP #1	Value
Response	0x0000 (Operation Successful)

Table 4.52: Network Protocol Filtering with 802.1Q Tag Header Packets, Phase I, Pass criteria (message)

The IUT (NAP/GN) interprets the IEEE 802.1Q Tag header and forwards only test patterns 12, 13, 14, 16, 17, and 18 to Tester2 (PANU). The packages sent to Tester2 (PANU) contain the data below:

Pattern #n	Value
L2CAP Length	0x5EF (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	Tester2 (PANU)'s Bluetooth Address
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x8100 (IEEE 802.1Q Tag header)
Data [0-1]	0xXXXX (Device MAY interpret the 802.1Q Tag Header)
Data [2-3]	Networking Protocol Type depending on which test pattern: Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01,…, 0x9E, 0x9F}

Table 4.53: Network Protocol Filtering with 802.1Q Tag Header Packets, Phase I, Pass criteria (data, option 1)

Or

Pattern #n	Value
L2CAP Length	0x5E9 (BNEP packet length within the L2CAP packet)
BNEP Type	0x03 (BNEP_COMPRESSED_ETHERNET_SOURCE_ONLY)
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x8100 (IEEE 802.1Q Tag header)
Data [0-1]	0xXXXX (Device MAY interpret the 802.1Q Tag header)
Data [2-3]	Networking Protocol Type depending on which test pattern: Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8
Dependent Header or other payload information	60 bytes of data to be ignored

Pattern #n	Value
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.54: Network Protocol Filtering with 802.1Q Tag Header Packets, Phase I, Pass criteria (data, option 2)

Phase II

The IUT (NAP/GN) receives and interprets the BNEP_FILTER_NET_TYPE_SET_MSG from Tester2 (PANU) and responds to that correctly with the following message:

CTRL Msg RSP #1	Value
L2CAP Length	0x0004 (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (BNEP Extension flag = no extension header)
BNEP Control Type	0x04 (BNEP_FILTER_NET_TYPE_RESPONSE_MSG)
Response	0x0000 (Operation Successful)

Table 4.55: Network Protocol Filtering with 802.1Q Tag Header Packets, Phase II, Pass criteria (message)

The IUT (NAP/GN) interprets the IEEE 802.1Q Tag header and forwards all of the test patterns #10 – #19 to Tester2 (PANU). The packages sent to Tester2 (PANU) contain the data below:

Pattern #n	Value
L2CAP Length	0x5EF (BNEP packet length within the L2CAP packet)
BNEP Type	0x00 (BNEP_GENERAL_ETHERNET)
Destination Address (48 bits)	Tester2 (PANU)'s Bluetooth Address
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x8100 (IEEE 802.1Q Tag header)
Data [0-1]	0xXXXX (Device MAY interpret the 802.1Q Tag header)
Data [2-3]	Networking Protocol Type depending on which test pattern: Pattern 10: Base Networking Protocol Type + 0 Pattern 11: Base Networking Protocol Type + 1 Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 15: Base Networking Protocol Type + 5 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8 Pattern 19: Base Networking Protocol Type + 9
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01 , …, 0x9E, 0x9F}

Table 4.56: Network Protocol Filtering with 802.1Q Tag Header Packets, Phase II, Pass criteria (data, option 1)

Or

Pattern #n	Value
L2CAP Length	0x5E9 (BNEP packet length within the L2CAP packet)
BNEP Type	0x03
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x8100 (IEEE 802.1Q Tag header)
Data [0-1]	0xXXXX (Device MAY interpret the 802.1Q Tag header)
Data [2-3]	Networking Protocol Type depending on which test pattern: Pattern 10: Base Networking Protocol Type + 0 Pattern 11: Base Networking Protocol Type + 1 Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 15: Base Networking Protocol Type + 5 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8 Pattern 19: Base Networking Protocol Type + 9
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x5A0 bytes are: {0x00, 0x01,…,0xFF} *5 + {0x00, 0x01,…, 0x9E, 0x9F}

Table 4.57: Network Protocol Filtering with 802.1Q Tag Header Packets, Phase II, Pass criteria (data, option 2)

4.6.4 Sending Unknown Extension and Setting Multicast Address Filter

- Test Purpose

  Verify the correct implementation of BNEP_FILTER_MULTI_ADDR_SET_MSG. The IUT (NAP/GN), which is piconet central, is the target device of this test.

  This test involves three Bluetooth devices that form one piconet. Thus, two testers (PANUs) (peripherals) are connected to the IUT (NAP/GN) that is the piconet central. One of the PANUs, Tester2, will set a Multicast Address Filter in the IUT (NAP/GN).

  Tester1 will send several packets (containing two unknown extension headers and data payload) to the IUT (NAP/GN). The IUT (NAP/GN) receives the packets from the originator and forwards those packets (including the data payload and the unknown extension headers) to the destination node, if they have not been filtered out. If they are filtered out, the Ethernet payload will be dropped, the protocol type will be set to 0, and the unknown extension headers will be forwarded.

- Reference

  BNEP [1] and L2CAP in [2]

- Initial Condition

  - Tester1 (PANU), Tester2 (PANU), and the IUT (NAP/GN) must be switched on and initialized.
  - Initialization may require entering a PIN if there are any security restrictions already in place on the Bluetooth link.
  - Tester1 (PANU), Tester2 (PANU), and the IUT (NAP/GN) must have valid Bluetooth device addresses (BD_ADDR). The IUT (NAP/GN) must have an open and configured BNEP channel with Tester1 (PANU) and with Tester2 (PANU).

- Tester1 (PANU) and Tester2 (PANU) must not have any direct connection with each other. Tester1 (PANU) must know about the BD_ADDR of Tester2 (PANU).
- For this test, a base multicast address must be chosen and must be a valid multicast address (for example, 0x030002300000); all test patterns use this base address plus an integer offset to determine which multicast address to use in each test pattern. For example, if a multicast address of 0x030002300000 is chosen as the base address, then if a test pattern states a value of Base Address + 2, then the value for the multicast address for the test pattern will be 0x030002300002.

- Test Case Configuration

Test Case
PAN/NAP/BNEP/FILTER/BV-13-C [Sending Unknown Extension and Setting Multicast Address Filter]
PAN/GN/BNEP/FILTER/BV-13-C [Sending Unknown Extension and Setting Multicast Address Filter]

Table 4.58: Sending Unknown Extension and Setting Multicast Address Filter test cases

- Test Procedure

  In Phase I, the IUT (NAP/GN) receives a BNEP_FILTER_MULTI_ADDR_SET_MSG containing two filter ranges from Tester2 (PANU) and responds to this command by sending a BNEP_FILTER_MULTI_ADDR_RESPONSE_MSG back to Tester2 (PANU). If the response is positive, Tester1 (PANU) starts sending the test pattern #0 – #9 to the IUT (NAP/GN), which has to forward to Tester2 (PANU) based on the filter settings.

  In Phase II, the IUT (NAP/GN) receives a BNEP_FILTER_MULTI_ADDR_SET_MSG, which resets the filter ranges from Tester2 (PANU) and responds to this command by sending a BNEP_FILTER_MULTI_ADDR_RESPONSE_MSG back to Tester2 (PANU). If the response is positive, Tester1 (PANU) starts sending the test pattern #0 – #9 to the IUT (NAP/GN), which has to forward to Tester2 (PANU) based on the reset filter settings.

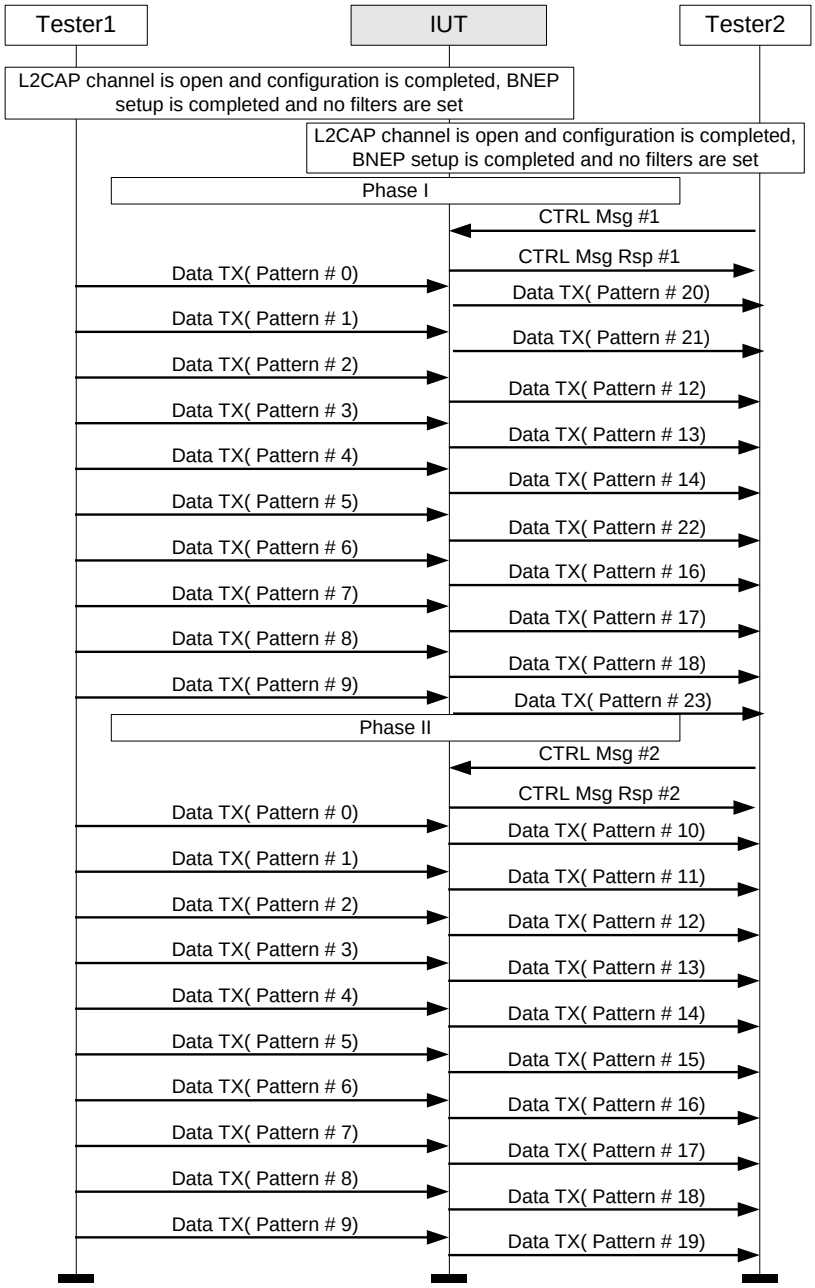

Figure 4.13: Sending Unknown Extension and Setting Multicast Address Filter

Phase I

Tester2 (PANU) sends the command below:

CTRL Msg #1	**Value**
L2CAP Length	0x001C (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (No extension header)
BNEP Control Type	0x05 (BNEP_FILTER_MULTI_ADDR_SET_MSG)
List Length	0x0018 (Byte length of the list, of the multicast address not to be filtered)

CTRL Msg #1	Value
Multicast Address Range Start #1	Multicast address range starting at Base Address + 2
Multicast Address Range End #1	Multicast address range ending at Base Address + 4
Multicast Address Range Start #2	Multicast address range starting at Base Address + 6
Multicast Address Range End #2	Multicast address range ending at Base Address + 8

Table 4.59: Sending Unknown Extension and Setting Multicast Address Filter, Phase I, Tester2 command

Tester1 (PANU) sends the patterns below:

Pattern #0–9	Value
L2CAP Length	0x069B (BNEP packet length within the L2CAP packet, equal to BNEP minimum MTU = 1691)
BNEP Type + Extension	0x80 BNEP_GENERAL_ETHERNET = 0x00 (7 bits), Extension=0x1 (1 bit)
Destination Address	Multicast address depending on which test pattern: Pattern 0: Base Address + 0 Pattern 1: Base Address + 1 Pattern 2: Base Address + 2 Pattern 3: Base Address + 3 Pattern 4: Base Address + 4 Pattern 5: Base Address + 5 Pattern 6: Base Address + 6 Pattern 7: Base Address + 7 Pattern 8: Base Address + 8 Pattern 9: Base Address + 9
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x0800, or 0x86DD, or 0xXXXX
Extension header + Extension	0xD5 Extension header: 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF (Extension length = 255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01.0xFE)
Extension header + Extension	0x7F Extension header: 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF Extension length = 255
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF, 0xFE, .0x01)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x44E (1102) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x4C, 0x4D}

Table 4.60: Sending Unknown Extension and Setting Multicast Address Filter, Phase I, Tester1 patterns

Phase II

Tester2 (PANU) sends the command below:

CTRL Msg #2	Value
L2CAP Length	0x0004 (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (No extension header)
BNEP Control Type	0x05 (BNEP_FILTER_MULTI_ADDR_SET_MSG)
List Length	0x0000 (Length of the list, in bytes, of the multicast addresses not to be filtered. Length = 0 = reset filters)

Table 4.61: Sending Unknown Extension and Setting Multicast Address Filter, Phase II, Tester2 command

Tester1 (PANU) sends the patterns below:

Pattern #0–9	Value
L2CAP Length	0x069B (BNEP packet length within the L2CAP packet, equal to the BNEP minimum MTU = 1691 bytes)
BNEP Type + Extension	0x80 BNEP Type field is: BNEP_GENERAL_ETHERNET = 0x00 (7 bits) Extension = 0x1 (1 bit)
Destination Address	Multicast address depending on which test pattern: Pattern 0: Base Address + 0 Pattern 1: Base Address + 1 Pattern 2: Base Address + 2 Pattern 3: Base Address + 3 Pattern 4: Base Address + 4 Pattern 5: Base Address + 5 Pattern 6: Base Address + 6 Pattern 7: Base Address + 7 Pattern 8: Base Address + 8 Pattern 9: Base Address + 9
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Extension header + Extension	0xD5 Extension header is 0x55 (7 bits) Extension = 0x1 (1 bit)
Extension Length	0xFF Extension length = 255
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01.0xFE)
Extension header + Extension	0x7F Extension header: 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF (Extension length = 255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF, 0xFE, .0x01)
Dependent Header or other payload information	60 bytes of data to be ignored

Pattern #0–9	Value
Data	0x44E (1102) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x4C, 0x4D}

Table 4.62: Sending Unknown Extension and Setting Multicast Address Filter, Phase II, Tester1 patterns

- Expected Outcome

  Pass verdict

  Phase I

  The IUT (NAP/GN) receives and interprets the BNEP_FILTER_MULTI_ADDR_SET_MSG from Tester2 (PANU) and responds to that correctly with the following message:

CTRL Msg RSP #1	Value
L2CAP Length	0x0004 (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (No extension header)
BNEP Control Type	0x06 (BNEP_FILTER_MULTI_ADDR_RESPONSE_MSG)
Response	0x0000 (Operation Successful)

Table 4.63: Sending Unknown Extension and Setting Multicast Address Filter, Phase I, Pass criteria (message)

The IUT (NAP/GN) forwards test patterns 12, 13, 14, 16, 17, and 18 to Tester2 (PANU), with the data payload. The packages sent to Tester2 (PANU) contain the data below:

Pattern #n	Value
L2CAP Length	0x069B (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x80 BNEP_GENERAL_ETHERNET = 0x00 (7 bits), Extension=0x1 (1 bit)
Destination Address	Multicast address depending on which test pattern: Pattern 12: Base Address + 2 Pattern 13: Base Address + 3 Pattern 14: Base Address + 4 Pattern 16: Base Address + 6 Pattern 17: Base Address + 7 Pattern 18: Base Address + 8
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Extension header + Extension	0xD5 Extension header is 0x55 (7 bits) Extension = 0x1 (1 bit)
Extension Length	0xFF Extension length = 255
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01..0xFE)
Extension header + Extension	0x7F Extension header is 0x7F (7 bits) Extension = 0x0 (1 bit)

Pattern #n	Value
Extension Length	0xFF
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF,0xFE..0x01)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x44E (1102) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x4C, 0x4D}

Table 4.64: Sending Unknown Extension and Setting Multicast Address Filter, Phase I, Pass criteria (data)

The IUT (NAP/GN) also forwards test pattern 20, 21, 22, and 23 to Tester2 (PANU), without the data payload, due to filtering. The packages sent to Tester2 (PANU) contain the data below:

Pattern #n	Value
L2CAP Length	0x0211 (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x80 BNEP_GENERAL_ETHERNET = 0x00 (7 bits), Extension = 0x1 (1 bit)
Destination Address	Multicast address depending on which test pattern: Pattern 20: Base Address + 0 Pattern 21: Base Address + 1 Pattern 22: Base Address + 5 Pattern 23: Base Address + 9
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x0000
Extension header + Extension	0xD5 Extension header is 0x55 (7 bits) Extension = 0x1 (1 bit)
Extension Length	0xFF
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01..0xFE)
Extension header + Extension	0x7F Extension header is 0x7F (7 bits) Extension = 0x0 (1 bit)
Extension Length	0xFF
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF,0xFE..0x01)

Table 4.65: Sending Unknown Extension and Setting Multicast Address Filter, Phase I, Pass criteria (data)

Phase II

The IUT (NAP/GN) receives and interprets the BNEP_FILTER_MULTI_ADDR_SET_MSG from Tester2 (PANU) and responds to that correctly with the following message:

CTRL Msg RSP #1	Value
L2CAP Length	0x0004 (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (No extension header)

CTRL Msg RSP #1	Value
BNEP Control Type	0x06 (BNEP_FILTER_MULTI_ADDR_RESPONSE_MSG)
Response	0x0000 (Operation Successful)

Table 4.66: Sending Unknown Extension and Setting Multicast Address Filter, Phase II, Pass criteria (message)

The IUT (NAP/GN) forwards all of the test patterns #10 – #19 to Tester2 (PANU). The packages sent to Tester2 (PANU) contain the data below:

Pattern #n	Value
L2CAP Length	0x069B (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x80 BNEP_GENERAL_ETHERNET = 0x00 (7 bits), Extension = 0x1 (1 bit)
Destination Address	Multicast address depending on which test pattern: Pattern 10: Base Address + 0 Pattern 11: Base Address + 1 Pattern 12: Base Address + 2 Pattern 13: Base Address + 3 Pattern 14: Base Address + 4 Pattern 15: Base Address + 5 Pattern 16: Base Address + 6 Pattern 17: Base Address + 7 Pattern 18: Base Address + 8 Pattern 19: Base Address + 9
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x0800 (IPv4), or 0x86DD (IPv6), or 0xXXXX (Other)
Extension header + Extension	0xD5 Extension header is 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF
Unknown Extension header payload	0xFF bytes, 255 bytes of payload which are (0x00,0x01..0xFE)
Extension header + Extension	0x7F
Extension Length	0xFF
Unknown Extension header payload	0xFF bytes, 255 bytes of payload which are (0xFF,0xFE..0x01)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x44E (1102) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x4C, 0x4D}

Table 4.67: Sending Unknown Extension and Setting Multicast Address Filter, Phase II, Pass criteria (data)

4.6.5 Sending Unknown Extension and Setting Network Protocol Filter

- Test Purpose

  Verify the correct implementation of BNEP_FILTER_NET_TYPE_SET_MSG. The IUT (NAP/GN), which piconet central, is the target device of this test.

  This test involves three Bluetooth devices that form one piconet. Thus, two testers (PANUs) (peripherals) are connected to the IUT (NAP/GN) that is the piconet central. One of the PANUs, Tester2, will set a Network Protocol Address Filter in the IUT (NAP/GN). The other PANU will send several packets (containing two unknown extension headers and data payload) to the IUT (NAP/GN). The IUT (NAP/GN) receives the packets from the originator and forwards those packets (including the data payload and the unknown extension headers) to the destination node, if they have not been filtered out. If they are filtered out, the Ethernet payload will be dropped, the protocol type will be set to 0, and the unknown extension headers will be forwarded.

- Reference

  BNEP [1] and L2CAP in [2]

- Initial Condition

  - Tester1 (PANU), Tester2 (PANU), and the IUT (NAP/GN) must be switched on and initialized.
  - Initialization may require entering a PIN if there are any security restrictions already in place on the Bluetooth link.
  - Tester1 (PANU), Tester2 (PANU), and the IUT (NAP/GN) must have valid Bluetooth device addresses (BD_ADDR).
  - The IUT (NAP/GN) must have an open and configured BNEP channel with Tester1 (PANU) and with Tester2 (PANU).
  - Tester1 (PANU) and Tester2 (PANU) must not have any direct connection with each other.
  - Tester1 (PANU) must know about the BD_ADDR of Tester2 (PANU).
  - For this test, a base networking protocol type must be chosen and must be a networking protocol type (for example, 0x0800); all test patterns use this base networking protocol type plus an integer offset to determine which networking protocol type to use in each test pattern. For example, if a networking protocol type of 0x0800 is chosen as the base networking protocol type, then if a test pattern states a value of base Networking Protocol Type + 2, then the value for the networking protocol type for the test pattern will be 0x0802.

- Test Case Configuration

Test Case
PAN/NAP/BNEP/FILTER/BV-14-C [Sending Unknown Extension and Setting Network Protocol Filter]
PAN/GN/BNEP/FILTER/BV-14-C [Sending Unknown Extension and Setting Network Protocol Filter]

Table 4.68: Sending Unknown Extension and Setting Network Protocol Filter test cases

- Test Procedure

  In Phase I, the IUT (NAP/GN) receives a BNEP_FILTER_NET_TYPE_SET_MSG containing two filter ranges from Tester2 (PANU) and responds to this command by sending a BNEP_FILTER_NET_TYPE_RESPONSE_MSG back to Tester2 (PANU). If the response is positive,

Tester1 (PANU) starts sending the test pattern #0 – #9 to the IUT (NAP/GN), which has to forward to Tester2 (PANU) based on the filter settings.

In Phase II, the IUT (NAP/GN) receives a BNEP_FILTER_NET_TYPE_SET_MSG, which resets the filter ranges from Tester2 (PANU) and responds to this command by sending a BNEP_FILTER_NET_TYPE_RESPONSE_MSG back to Tester2 (PANU). If the response is positive, Tester1 (PANU) starts sending the test pattern #0 – #9 to the IUT (NAP/GN), which has to forward to Tester2 (PANU) based on the reset filter settings.

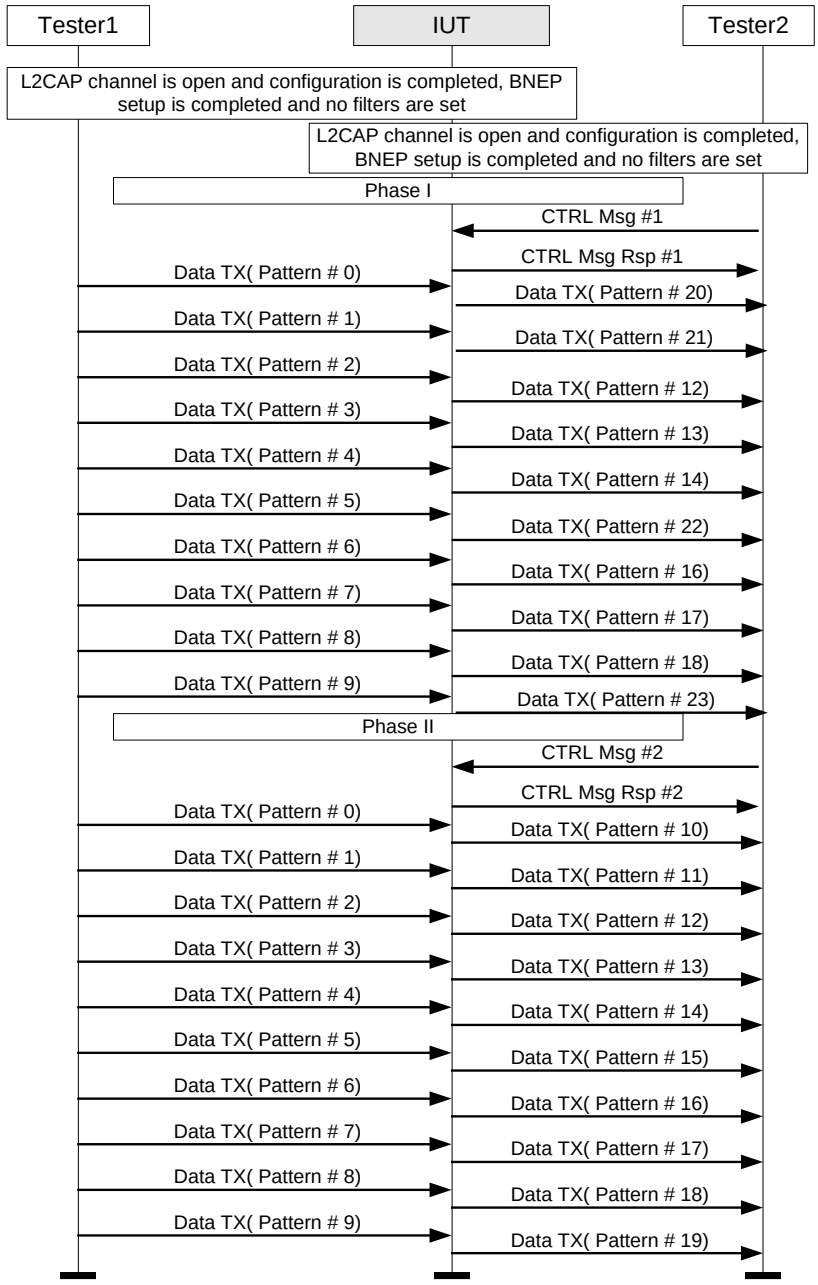

Figure 4.14: Sending Unknown Extension and Setting Network Protocol Filter

Phase I

Tester2 (PANU) sends the command below:

CTRL Msg #1	Value
L2CAP Length	0x000C (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (No extension header)
BNEP Control Type	0x03 (BNEP_FILTER_NET_TYPE_SET_MSG)
List Length	0x0008 Length of the list, in bytes, of the network protocols not to be filtered
Network Protocol Type Range Start #1	Networking Protocol Type range starting at Base Networking Protocol Type + 2
Network Protocol Type Range End #1	Networking Protocol Type range starting at Base Networking Protocol Type + 4
Network Protocol Type Range Start #2	Networking Protocol Type range starting at Base Networking Protocol Type + 6
Network Protocol Type Range End #2	Networking Protocol Type range starting at Base Networking Protocol Type + 8

Table 4.69: Sending Unknown Extension and Setting Network Protocol Filter, Phase I, Tester2 command

Tester1 (PANU) sends the test patterns below:

Pattern #0–9	Value
L2CAP Length	0x069B (BNEP packet length within the L2CAP packet, equal to the BNEP minimum MTU = 1691 bytes)
BNEP Type + Extension	0x80
Destination Address (48 bits)	Tester2 (PANU)'s Bluetooth Address
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	Networking Protocol Type depending on which test pattern: Pattern 0: Base Networking Protocol Type + 0 Pattern 1: Base Networking Protocol Type + 1 Pattern 2: Base Networking Protocol Type + 2 Pattern 3: Base Networking Protocol Type + 3 Pattern 4: Base Networking Protocol Type + 4 Pattern 5: Base Networking Protocol Type + 5 Pattern 6: Base Networking Protocol Type + 6 Pattern 7: Base Networking Protocol Type + 7 Pattern 8: Base Networking Protocol Type + 8 Pattern 9: Base Networking Protocol Type + 9
Extension header + Extension	0xD5 Extension header: 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes (255 bytes of payload which are (0x00,0x01.0xFE))
Extension header + Extension	0x7F (Extension header: 0x7F (7 bits), Extension = 0x0 (1 bit))
Extension Length	0xFF (255)

Pattern #0–9	Value
Unknown Extension header payload	0xFF bytes (255 bytes of payload which are (0xFF, 0xFE, .0x01))
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x44E (1102) bytes are:: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x4C, 0x4D}

Table 4.70: Sending Unknown Extension and Setting Network Protocol Filter, Phase I, Tester1 test patterns

Phase II

Tester2 (PANU) sends the command below:

CTRL Msg #2	Value
L2CAP Length	0x0004 (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (BNEP Extension flag = no extension header)
BNEP Control Type	0x03 (BNEP_FILTER_NET_TYPE_SET_MSG)
List Length	0x0000 (Length of the list, in bytes, of the network protocols not to be filtered. Length = 0 = Reset filters)

Table 4.71: Sending Unknown Extension and Setting Network Protocol Filter, Phase II, Tester2 command

Tester1 (PANU) sends the test patterns below:

Pattern #0–9	Value
L2CAP Length	0x069B (BNEP packet length within the L2CAP packet, equal to the BNEP minimum MTU = 1691 bytes)
BNEP Type + Extension	0x80 BNEP_GENERAL_ETHERNET = 0x00 (7 bits), Extension=0x1 (1 bit)
Destination Address (48 bits)	Tester2 (PANU)'s Bluetooth Address
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	Networking Protocol Type depending on which test pattern: Pattern 0: Base Networking Protocol Type + 0 Pattern 1: Base Networking Protocol Type + 1 Pattern 2: Base Networking Protocol Type + 2 Pattern 3: Base Networking Protocol Type + 3 Pattern 4: Base Networking Protocol Type + 4 Pattern 5: Base Networking Protocol Type + 5 Pattern 6: Base Networking Protocol Type + 6 Pattern 7: Base Networking Protocol Type + 7 Pattern 8: Base Networking Protocol Type + 8 Pattern 9: Base Networking Protocol Type + 9
Extension header + Extension	0xD5 Extension header: 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF (255)

Pattern #0–9	Value
Unknown Extension header payload	0xFF bytes (255 bytes of payload which are (0x00,0x01.0xFE))
Extension header + Extension	0x7F (Extension header: 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes (255 bytes of payload which are (0xFF, 0xFE,….0x01))
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x44E (1102) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x4C, 0x4D}

Table 4.72: Sending Unknown Extension and Setting Network Protocol Filter, Phase II, Tester1 test patterns

- Expected Outcome

  Pass verdict

  Phase I

  The IUT (NAP/GN) receives and interprets the BNEP_FILTER_NET_TYPE_SET_MSG from Tester2 (PANU) and responds to that correctly with the following message:

CTRL Msg RSP #1	Value
L2CAP Length	0x0004 (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (No extension header)
BNEP Control Type	0x04 (BNEP_FILTER_NET_TYPE_RESPONSE_MSG)
Response	0x0000 (Operation Successful)

Table 4.73: Sending Unknown Extension and Setting Network Protocol Filter, Phase I, Pass criteria (message)

The IUT (NAP/GN) forwards test patterns 12, 13, 14, 16, 17, and 18 to Tester2 (PANU). The packages sent to Tester2 (PANU) contains the data below:

Pattern #n	Value
L2CAP Length	0x069B (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x80 (BNEP_GENERAL_ETHERNET = 0x00 (7 bits), Extension = 0x1 (1 bit))
Destination Address (48 bits)	Tester2 (PANU)’s Bluetooth Address
Source Address (48 bits)	Tester1 (PANU)’s Bluetooth Address
Network Protocol Type	Networking Protocol Type depending on which test pattern: Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8

Pattern #n	Value
Extension header + Extension	0xD5 (Extension header is 0x55 (7 bits), Extension = 0x1 (1 bit))
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes (255 bytes of payload which are (0x00,0x01..0xFE))
Extension header + Extension	0x7F (Extension header: 0x7F (7 bits), Extension = 0x0 (1 bit))
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes (255 bytes of payload which are (0xFF,0xFE..0x01))
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x44E (1102) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x4C, 0x4D}

Table 4.74: Sending Unknown Extension and Setting Network Protocol Filter, Phase I, Pass criteria (data, Patterns 12, 13, 14, 16, 17, 18, option 1)

Or

Pattern #n	Value
L2CAP Length	0x0697 (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x83 BNEP_COMPRESSED_ETHERNET_SOURCE_ONLY = 0x03 (7 bits), Extension = 0x1 (1 bit)
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	Networking Protocol Type depending on which test pattern: Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8
Extension header + Extension	0xD5 Extension header: 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01..0xFE)
Extension header + Extension	0x7F Extension header: 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes (255 bytes of payload which are (0xFF,0xFE..0x01))
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x44E (1102) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x4C, 0x4D}

Table 4.75: Sending Unknown Extension and Setting Network Protocol Filter, Phase I, Pass criteria (data, Patterns 12, 13, 14, 16, 17, 18, option 2)

The IUT (NAP/GN) also forwards test patterns 20, 21, 22, and 23 to Tester2 (PANU), without the data payload, due to filtering. The packages sent to Tester2 (PANU) contain the data below:

Pattern #n	**Value**
L2CAP Length	0x0211 (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x80
Destination Address (48 bits)	Tester2 (PANU)'s Bluetooth Address
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x0000 (Networking Protocol Type is set to zero due to being filtered)
Extension header + Extension	0xD5 Extension header is 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes (255 bytes of payload which are (0x00,0x01..0xFE))
Extension header + Extension	0x7F (Extension header is 0x7F (7 bits), Extension = 0x0 (1 bit))
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF,0xFE..0x01)

Table 4.76: Sending Unknown Extension and Setting Network Protocol Filter, Phase I, Pass criteria (data, Patterns 20–23, option 1)

Or

Pattern #n	**Value**
L2CAP Length	0x020B (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x83 BNEP_COMPRESSED_ETHERNET_SOURCE_ONLY = 0x03 (7 bits), Extension = 0x1 (1 bit)
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x0000 Networking Protocol Type is set to zero due to being filtered
Extension header + Extension	0xD5 Extension header: 0x55 (7 bits) Extension = 0x1 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes
Extension header + Extension	0x7F Extension header: 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF,0xFE..0x01)

Table 4.77: Sending Unknown Extension and Setting Network Protocol Filter, Phase I, Pass criteria (data, Patterns 20–23, option 2)

Phase II

The IUT (NAP/GN) receives and interprets the BNEP_FILTER_NET_TYPE_SET_MSG from Tester2 (PANU) and responds to that correctly with the following message:

CTRL Msg RSP #1	Value	Description
L2CAP Length	0x0004	
BNEP Type	0x01	BNEP_CONTROL
Extension	0x0	BNEP Extension flag = 0x0 = no extension header
BNEP Control Type	0x04	BNEP_FILTER_NET_TYPE_RESPONSE_MSG
Response	0x0000	Operation Successful

Table 4.78: Sending Unknown Extension and Setting Network Protocol Filter, Phase II, Pass criteria (message)

The IUT (NAP/GN) forwards all of the test patterns #10 – #19 to Tester2 (PANU). The packages sent to Tester2 (PANU) contain the data below:

Pattern #n	Value
L2CAP Length	0x069B (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x80 BNEP_GENERAL_ETHERNET = 0x00 (7 bits) Extension = 0x1 (1 bit)
Destination Address (48 bits)	Tester2 (PANU)'s Bluetooth Address
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	Networking Protocol Type depending on which test pattern: Pattern 10: Base Networking Protocol Type + 0 Pattern 11: Base Networking Protocol Type + 1 Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 15: Base Networking Protocol Type + 5 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8 Pattern 19: Base Networking Protocol Type + 9
Extension header + Extension	0xD5 Extension header is 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01..0xFE)
Extension header + Extension	0x7F Extension header is 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF,0xFE..0x01)
Dependent Header or other payload information	60 bytes of data to be ignored

Pattern #n	Value
Data	0x44E (1102) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x4C, 0x4D}

Table 4.79: Sending Unknown Extension and Setting Network Protocol Filter, Phase II, Pass criteria (data, Patterns 10–19, option 1)

Or

Pattern #n	Value
L2CAP Length	0x0697 (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x83 BNEP_COMPRESSED_ETHERNET_SOURCE_ONLY = 0x03 (7 bits), Extension = 0x1 (1 bit)
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	Networking Protocol Type depending on which test pattern: Pattern 10: Base Networking Protocol Type + 0 Pattern 11: Base Networking Protocol Type + 1 Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 15: Base Networking Protocol Type + 5 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8 Pattern 19: Base Networking Protocol Type + 9
Extension header + Extension	0xD5 Extension header is 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01..0xFE)
Extension header + Extension	0x7F Extension header is 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF,0xFE..0x01)
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x44E (1102) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x4C, 0x4D}

Table 4.80: Sending Unknown Extension and Setting Network Protocol Filter, Phase II, Pass criteria (data, Patterns 10–19, option 2)

4.6.6 Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets

- Test Purpose

  Verify the correct implementation of network protocol filtering with 802.1Q tag header packets. The IUT (NAP/GN), which piconet central, is the target device of this test.

  This test involves three Bluetooth devices that form one piconet. Thus, two testers (PANUs) (peripherals) are connected to the IUT (NAP/GN) that is the piconet central. One of the PANUs, Tester2, will set a Network Protocol Address Filter in the IUT (NAP/GN). The other PANU will send several packets (containing two unknown extension headers and data payload) to the IUT (NAP/GN). The IUT (NAP/GN) receives the packets from the originator and forwards those packets (including the data payload and the unknown extension headers) to the destination node, if they have not been filtered out. If they are filtered out, the Ethernet payload will be dropped, the actual packet length will be set to zero, and the unknown extension headers will be forwarded. The Network Protocol Type in the BNEP header is unchanged from the value indicating an 802.1Q header, and the 802.1Q Tag Control Information is also unchanged and forwarded together with the new packet length.

- Reference

  BNEP [1] and L2CAP in [2]

- Initial Condition

  - Tester1 (PANU), Tester2 (PANU), and the IUT (NAP/GN) must be switched on and initialized.
  - Initialization may require entering a PIN if there are any security restrictions already in place on the Bluetooth link.
  - Tester1 (PANU), Tester2 (PANU), and the IUT (NAP/GN) must have valid Bluetooth device addresses (BD_ADDR).
  - The IUT (NAP/GN) must have an open and configured BNEP channel with Tester1 (PANU) and with Tester2 (PANU).
  - Tester1 (PANU) and Tester2 (PANU) must not have any direct connection with each other. Tester1 (PANU) must know about the BD_ADDR of Tester2 (PANU).
  - For this test, a base networking protocol type must be chosen and must be a networking protocol type (for example, 0x0800); all test patterns use this base networking protocol type plus an integer offset to determine which networking protocol type to use in each test pattern. For example, if a networking protocol type of 0x0800 is chosen as the base networking protocol type, then if a test pattern states a value of base Networking Protocol Type + 2, then the value for the networking protocol type for the test pattern will be 0x0802.

- Test Case Configuration

Test Case
PAN/NAP/BNEP/FILTER/BV-15-C [Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets]
PAN/GN/BNEP/FILTER/BV-15-C [Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets]

Table 4.81: Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets test cases

- Test Procedure

  In Phase I, the IUT (NAP/GN) receives a BNEP_FILTER_NET_TYPE_SET_MSG containing two filter ranges from Tester2 (PANU) and responds to this command by sending a BNEP_FILTER_NET_TYPE_RESPONSE_MSG back to Tester2 (PANU). If the response is positive, Tester1 (PANU) starts sending the test pattern #0 – #9 to the IUT (NAP/GN), which has to forward to Tester2 (PANU) based on the filter settings.

  In Phase II, the IUT (NAP/GN) receives a BNEP_FILTER_NET_TYPE_SET_MSG, which resets the filter ranges from Tester2 (PANU) and responds to this command by sending a BNEP_FILTER_NET_TYPE_RESPONSE_MSG back to Tester2 (PANU). If the response is positive, Tester1 (PANU) starts sending the test pattern #0 – #9 to the IUT (NAP/GN), which has to forward to Tester2 (PANU) based on the reset filter settings.

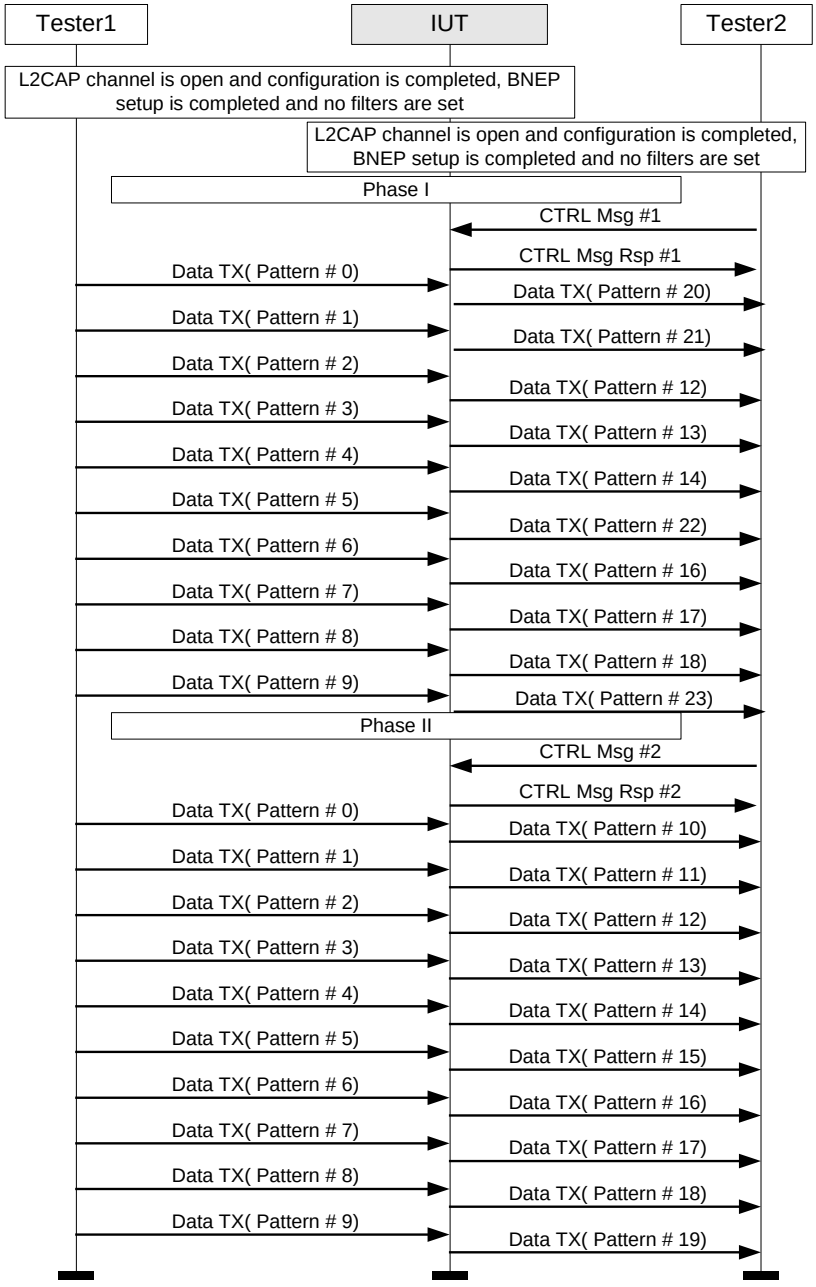

Figure 4.15: Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets

Phase I

Tester2 (PANU) sends the command below:

CTRL Msg#1	Value
L2CAP Length	0x000C (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (No extension header)
BNEP Control Type	0x03 (BNEP_FILTER_NET_TYPE_SET_MSG)
List Length	0x0008 Length of the list, in bytes, of the network protocols not to be filtered
Network Protocol Type Range Start #1	Base Networking Protocol Type + 2
Network Protocol Type Range End #1	Base Networking Protocol Type + 4
Network Protocol Type Range Start #2	Base Networking Protocol Type + 6
Network Protocol Type Range End #2	Base Networking Protocol Type + 8

Table 4.82: Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets, Phase I, Tester2 command

Tester1 (PANU) sends the test patterns below:

Pattern #0–9	Value
L2CAP Length	0x069B (BNEP packet length within the L2CAP packet, equal to the BNEP minimum MTU = 1691 bytes)
BNEP Type + Extension	0x80
Destination Address (48 bits)	Tester2 (PANU)'s Bluetooth Address
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x8100 Networking Protocol Type for IEEE 802.1Q tag header
Extension header + Extension	0xD5 Extension header is 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01,....0xFE)
Extension header + Extension	0x7F
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF, 0xFE,...,0x01)
Data [0-1]	0xXXXX (Device MAY interpret the 802.1Q tag header)

Pattern #0–9	Value
Data [2-3]	Networking Protocol Type depending on which test pattern: Pattern 0: Base Networking Protocol Type + 0 Pattern 1: Base Networking Protocol Type + 1 Pattern 2: Base Networking Protocol Type + 2 Pattern 3: Base Networking Protocol Type + 3 Pattern 4: Base Networking Protocol Type + 4 Pattern 5: Base Networking Protocol Type + 5 Pattern 6: Base Networking Protocol Type + 6 Pattern 7: Base Networking Protocol Type + 7 Pattern 8: Base Networking Protocol Type + 8 Pattern 9: Base Networking Protocol Type + 9
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x44A (1098) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x48, 0x49}

Table 4.83: Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets, Phase I, Tester1 test patterns

Phase II

Tester2 (PANU) sends the command below:

CTRL Msg #2	Value
L2CAP Length	0x0004 (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (No extension header)
BNEP Control Type	0x03 (BNEP_FILTER_NET_TYPE_SET_MSG)
List Length	0x0000 (Reset filters)

Table 4.84: Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets, Phase II, Tester2 command

Tester1 (PANU) sends the test patterns below:

Pattern #0–9	Value
L2CAP Length	0x069B (BNEP packet length within the L2CAP packet, equal to the BNEP minimum MTU = 1691 bytes)
BNEP Type + Extension	0x80 BNEP_GENERAL_ETHERNET: 0x00 (7 bits), Extension=0x1 (1 bit)
Destination Address (48 bits)	Tester2 (PANU)'s Bluetooth Address
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x8100 (IEEE 802.1Q tag header)
Extension header + Extension	0xD5 (Extension header is 0x55 (7 bits), Extension = 0x1 (1 bit))
Extension Length	0xFF (255)

Pattern #0–9	Value
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01,...0xFE)
Extension header + Extension	0x7F Extension header is 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF, 0xFE,...0x01)
Data [0-1]	0xXXXX (IEEE 802.1Q tag header)
Data [2-3]	Networking Protocol Type depending on which test pattern: Pattern 0: Base Networking Protocol Type + 0 Pattern 1: Base Networking Protocol Type + 1 Pattern 2: Base Networking Protocol Type + 2 Pattern 3: Base Networking Protocol Type + 3 Pattern 4: Base Networking Protocol Type + 4 Pattern 5: Base Networking Protocol Type + 5 Pattern 6: Base Networking Protocol Type + 6 Pattern 7: Base Networking Protocol Type + 7 Pattern 8: Base Networking Protocol Type + 8 Pattern 9: Base Networking Protocol Type + 9
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x44A (1098) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x48, 0x49}

Table 4.85: Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets, Phase II, Tester1 test patterns

- Expected Outcome

  Pass verdict

  Phase I

  The IUT (NAP/GN) receives and interprets the BNEP_FILTER_NET_TYPE_SET_MSG from Tester2 (PANU) and responds to that correctly with the following message:

CTRL Msg RSP #1	Value	Description
L2CAP Length	0x0004	
BNEP Type	0x01	BNEP_CONTROL
Extension	0x0	BNEP Extension flag = 0x0 = no extension header
BNEP Control Type	0x04	BNEP_FILTER_NET_TYPE_RESPONSE_MSG
Response	0x0000	Operation Successful

Table 4.86: Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets, Phase I, Pass criteria (message)

The IUT (NAP/GN) interprets the IEEE 802.1Q Tag header and forwards test patterns 12, 13, 14, 16, 17, and 18 to Tester2 (PANU). The packages sent to Tester2 (PANU) contain the data below:

Pattern #n	**Value**
L2CAP Length	0x069B (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x80 BNEP_GENERAL_ETHERNET = 0x00 (7 bits) Extension = 0x1 (1 bit)
Destination Address (48 bits)	Tester2 (PANU)'s Bluetooth Address
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x8100 (IEEE 802.1Q Tag header)
Extension header + Extension	0xD5
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01,...0xFE)
Extension header + Extension	0x7F Extension header is 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF, 0xFE,...0x01)
Data [0-1]	0xXXXX (IEEE 802.1Q tag header)
Data [2-3]	Networking Protocol Type depending on which test pattern: Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x44A (1098) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x48, 0x49}

Table 4.87: Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets, Phase I, Pass criteria (data, Patterns 12, 13, 14, 16, 17, 18, option 1)

Or

Pattern #n	**Value**
L2CAP Length	0x0697 (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x83 BNEP_COMPRESSED_ETHERNET_SOURCE_ONLY = 0x03 (7 bits), Extension = 0x1 (1 bit)
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x8100 (IEEE 802.1Q Tag header)

Pattern #n	Value
Extension header + Extension	0xD5 Extension header is 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01,...0xFE)
Extension header + Extension	0x7F Extension header is 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF, 0xFE,...0x01)
Data [0-1]	0xXXXX (IEEE 802.1Q tag header)
Data [2-3]	Networking Protocol Type depending on which test pattern: Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x44A (1098) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x48, 0x49}

Table 4.88: Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets, Phase I, Pass criteria (data, Patterns 12, 13, 14, 16, 17, 18, option 2)

The IUT (NAP/GN) also forwards test patterns 20, 21, 22, and 23 to Tester2 (PANU), without the data payload, due to filtering. The packages sent to Tester2 (PANU) contain the data below:

Pattern #n	Value
L2CAP Length	0x0215 (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x80 BNEP_GENERAL_ETHERNET = 0x00 (7 bits) Extension = 0x1 (1 bit)
Destination Address (48 bits)	Tester2 (PANU)'s Bluetooth Address
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x8100 (IEEE 802.1Q Tag header)
Extension header + Extension	0xD5 Extension header is 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01,...0xFE)
Extension header + Extension	0x7F Extension header is 0x7F (7 bits), Extension = 0x0 (1 bit)

Pattern #n	Value
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF, 0xFE,….0x01)
Data [0-1]	0xXXXX (IEEE 802.1Q tag header)
Data [2-3]	0x0000 (Networking Protocol Type is set to zero due to the packet has been filtered)

Table 4.89: Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets, Phase I, Pass criteria (data, Patterns 20–23, option 1)

Or

Pattern #n	Value
L2CAP Length	0x020F (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x83 BNEP_COMPRESSED_ETHERNET_SOURCE_ONLY = 0x03 (7 bits), Extension = 0x1 (1 bit)
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x8100 (IEEE 802.1Q Tag header)
Extension header + Extension	0xD5 Extension header is 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01,...0xFE)
Extension header + Extension	0x7F Extension header is 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF, 0xFE,...0x01)
Data [0-1]	0xXXXX (IEEE 802.1Q tag header)
Data [2-3]	0x0000 (Networking Protocol Type is set to zero due to the packet has been filtered)

Table 4.90: Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets, Phase I, Pass criteria (data, Patterns 20–23, option 2)

Phase II

The IUT (NAP/GN) receives and interprets the BNEP_FILTER_NET_TYPE_SET_MSG from Tester2 (PANU) and responds to that correctly with the following message:

CTRL Msg RSP #1	Value
L2CAP Length	0x0004 (BNEP packet length within the L2CAP packet)
BNEP Type	0x01 (BNEP_CONTROL)
Extension	0x0 (No extension header)

CTRL Msg RSP #1	Value
BNEP Control Type	0x04 (BNEP_FILTER_NET_TYPE_RESPONSE_MSG)
Response	0x0000 (Operation Successful)

Table 4.91: Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets, Phase II, Pass criteria (message)

The IUT (NAP/GN) interprets the IEEE 802.1Q Tag header and forwards all of the test patterns #10 – #19 to Tester2 (PANU). The packages sent to Tester2 (PANU) contain the data below:

Pattern #n	Value
L2CAP Length	0x069B (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x80 BNEP_GENERAL_ETHERNET = 0x00 (7 bits), Extension = 0x1 (1 bit)
Destination Address (48 bits)	Tester2 (PANU)'s Bluetooth Address
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x8100 (IEEE 802.1Q Tag header)
Extension header + Extension	0xD5 (Extension header is 0x55 (7 bits) Extension = 0x1 (1 bit))
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01,...0xFE)
Extension header + Extension	0x7F Extension header is 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes (255 bytes of payload which are (0xFF, 0xFE,...0x01))
Data [0-1]	0xXXXX (IEEE 802.1Q tag header)
Data [2-3]	Networking Protocol Type depending on which test pattern: Pattern 10: Base Networking Protocol Type + 0 Pattern 11: Base Networking Protocol Type + 1 Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 15: Base Networking Protocol Type + 5 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8 Pattern 19: Base Networking Protocol Type + 9
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x44A (1098) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x48, 0x49}

Table 4.92: Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets, Phase II, Pass criteria (data, Patterns 10–19, option 1)

Or

Pattern #n	Value
L2CAP Length	0x069B (BNEP packet length within the L2CAP packet)
BNEP Type + Extension	0x83 BNEP_COMPRESSED_ETHERNET_SOURCE_ONLY = 0x03 (7 bits), Extension = 0x1 (1 bit)
Source Address (48 bits)	Tester1 (PANU)'s Bluetooth Address
Network Protocol Type	0x8100 (IEEE 802.1Q Tag header)
Extension header + Extension	0xD5 Extension header is 0x55 (7 bits), Extension = 0x1 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0x00,0x01,...0xFE)
Extension header + Extension	0x7F Extension header is 0x7F (7 bits), Extension = 0x0 (1 bit)
Extension Length	0xFF (255)
Unknown Extension header payload	0xFF bytes 255 bytes of payload which are (0xFF, 0xFE,...0x01)
Data [0-1]	0xXXXX (IEEE 802.1Q tag header)
Data [2-3]	Networking Protocol Type depending on which test pattern: Pattern 10: Base Networking Protocol Type + 0 Pattern 11: Base Networking Protocol Type + 1 Pattern 12: Base Networking Protocol Type + 2 Pattern 13: Base Networking Protocol Type + 3 Pattern 14: Base Networking Protocol Type + 4 Pattern 15: Base Networking Protocol Type + 5 Pattern 16: Base Networking Protocol Type + 6 Pattern 17: Base Networking Protocol Type + 7 Pattern 18: Base Networking Protocol Type + 8 Pattern 19: Base Networking Protocol Type + 9
Dependent Header or other payload information	60 bytes of data to be ignored
Data	0x44A (1098) bytes are: {0x00, 0x01,…,0xFF} * 3 + {0x00, 0x01 , …, 0x48, 0x49}

Table 4.93: Sending Unknown Extension and Network Protocol Filtering with 802.1Q Tag Header Packets, Phase II, Pass criteria (data, Patterns 10–19, option 2)

4.7 PAN Forwarding test cases

The PAN Forwarding test cases cover scenarios involving traffic passing between a Bluetooth PANU and some other network technology via a Network Access Point (NAP) as specified in the PAN profile.

PAN/NAP/BRIDGE/TX/BV-01-C [NAP Forwarding of a Unicast Packet from a Bluetooth PANU to other Network Technology e.g., Ethernet or GPRS]

- Test Purpose

  Verify that BNEP encapsulated IP packets sent by Tester1 (PANU) to Tester2 (Network Terminal) are converted to the correct format for the non-Bluetooth network and delivered to Tester2 (Network

Terminal) when any valid BNEP packet type is used. This test is carried out for all versions of IP supported by the IUT (NAP).

This test applies to Bluetooth NAPs (IUTs) that function as IP forwarders or routers between a PANU (Tester1) and a non-Bluetooth network infrastructure. Acting as testers are a Bluetooth PANU (Tester1) and a non-Bluetooth Network Terminal (Tester2). Tester2 (Network Terminal) is connected to the IUT (NAP) by a non-Bluetooth network infrastructure.

- Reference

  PAN Profile [5]

- Initial Condition

  - Tester1 (PANU) and the IUT (NAP) must have valid Bluetooth device addresses.
  - Both Tester1 (PANU) and the IUT (NAP) are switched on and initialized. Either Tester1 (PANU) or the IUT (NAP) has initiated the establishment and configuration of L2CAP and BNEP connections. No filters are set on the BNEP connection.
  - Tester1 (PANU) and Tester2 (Network Terminal) will have completed configuration of their IP stacks with valid IPv4 or IPv6 addresses to match the version of IP being tested in the IUT (NAP).
  - The connection procedure may require a PIN if there are any security restrictions already in place on the Bluetooth link. In addition, network connectivity has been established between Tester2 (Network Terminal) and the IUT (NAP).

- Test Procedure

  Tester2 (Network Terminal) listens for incoming packets and lists the packets it receives.

  Tester1 (PANU) sends a set of BNEP packets containing Unicast IP packets addressed to Tester2 (Network Terminal)'s IP address. These packets will vary in their BNEP type in order to exercise the IUT (NAP)'s ability to receive all valid BNEP packet types.

  As in the normal course of transmitting an IP packet, Tester1 (PANU) first determines the MAC address to which the BNEP packets will be sent. This will require knowledge of the default router address when the subnet mask indicates that the destination IP address is on a different IP subnet to Tester1 (PANU). When the destination MAC address is NOT the same as the IUT (NAP)'s BD_ADDR, two BNEP packets are sent, one of type BNEP_GENERAL_ETHERNET and one of type BNEP_COMPRESSED_DEST_ONLY. In all cases, the packet payloads contain a verifiable data sequence and are of a size equal to the BNEP MTU of 1500 bytes.

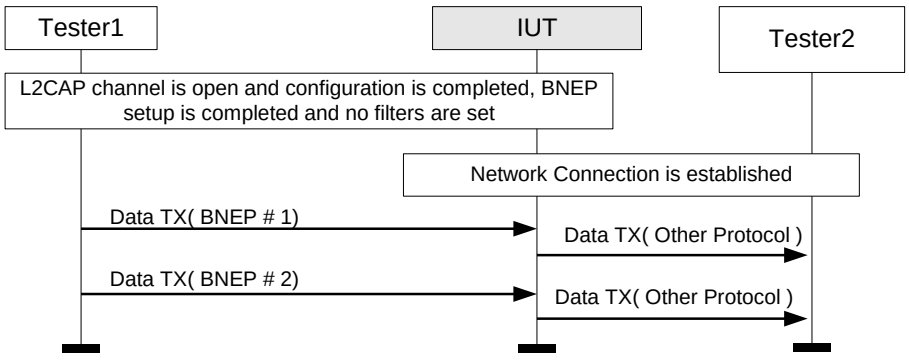

Figure 4.16: PAN/NAP/BRIDGE/TX/BV-01-C [NAP Forwarding of a Unicast Packet from a Bluetooth PANU to other Network Technology e.g. Ethernet or GPRS]

- Expected Outcome

  Pass verdict

  The packets sent by Tester1 (PANU) are correctly converted by the IUT (NAP) to the non-Bluetooth network format and forwarded or routed to Tester2 (Network Terminal). As proof of this and to verify the integrity of the data, Tester2 (Network Terminal) will list the packets it receives during the course of the test.

PAN/NAP/BRIDGE/RX/BV-02-C [NAP Forwarding of a Unicast Packet from a Non-Bluetooth Network to a Bluetooth PANU]

- Test Purpose

  This test applies to Bluetooth NAPs that function as IP forwarders or routers between a non-Bluetooth network infrastructure and Bluetooth PANUs. Supporting the test are a Bluetooth PANU and a non-Bluetooth Tester2 (Network Terminal). Tester2 (Network Terminal) is connected to the IUT (NAP) by a non-Bluetooth network infrastructure. The test will verify that the IUT (NAP) correctly receives and routes or forwards IP data packets sent by Tester2 (Network Terminal) to Tester1 (PANU). The correct processing of these packets requires that they are converted from the packet format of the non-Bluetooth network to valid BNEP packets before being delivered to Tester1 (PANU). This test is carried out for all versions of IP supported by the IUT (NAP).

- Reference

  PAN Profile [5]

- Initial Condition

  - Tester1 (PANU) and the IUT (NAP) must have valid Bluetooth device addresses.
  - Both Tester1 (PANU) and the IUT (NAP) are switched on and initialized.
  - Either Tester1 (PANU) or the IUT (NAP) has initiated the establishment and configuration of L2CAP and BNEP connections. No filters are set on the BNEP connection.
  - Tester1 (PANU) and Tester2 (Network Terminal) have completed configuration of their IP stacks with valid IPv4 or IPv6 addresses to match the version of IP being tested in the IUT (NAP).
  - The connection procedure may require a PIN if there are any security restrictions already in place on the Bluetooth link. In addition, network connectivity has been established between Tester2 (Network Terminal) and the IUT (NAP).

- Test Procedure

  Tester1 (PANU) listens for incoming packets and lists the packets it receives.

  Tester2 (Network Terminal) sends a Unicast IP packet addressed to Tester1 (PANU)’s IP address. The payload of the packet sent by Tester2 (Network Terminal) should be a verifiable data sequence and of a size equal to the MTU of the non-Bluetooth network.

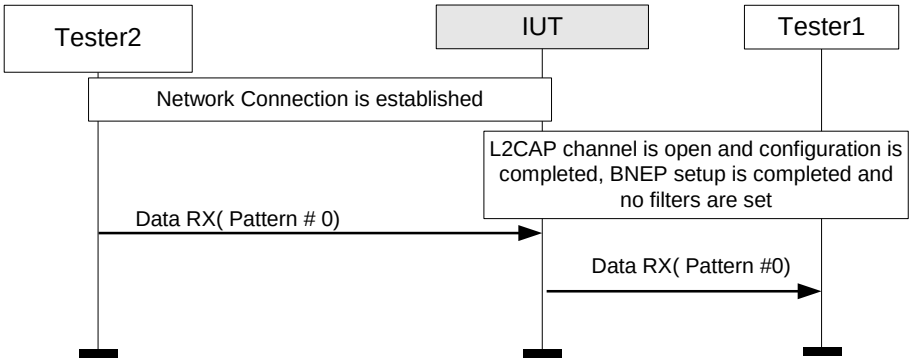

Figure 4.17: PAN/NAP/BRIDGE/RX/BV-02-C [NAP Forwarding of a Unicast Packet from a Non-Bluetooth Network to a Bluetooth PANU]

- Expected Outcome

  Pass verdict

  The IP packet sent by Tester2 (Network Terminal) reaches Tester1 (PANU) in a valid BNEP packet. The BNEP packet sent by the IUT (NAP) to Tester1 (PANU) may be any of the following packet types: BNEP_GENERAL_ETHERNET, BNEP_COMPRESSED_SRC_ONLY, BNEP_COMPRESSED, or BNEP_COMPRESSED_DEST_ONLY BNEP. As proof of the IUT (NAP)'s ability to route or forward the packet and to verify the integrity of the data, Tester1 (PANU) will list the packets it receives during the course of the test.

4.8 PAN IP Network test cases

This section describes the IP network test cases. The test cases, unless otherwise indicated, are valid for IPv4 as well as IPv6.

4.8.1 IP Address Assignment – Ad Hoc Networking (Autonet)

- Test Purpose

  Verify that an IUT can obtain a valid IP address in an ad hoc networking topology, as described in [4]. For NAP IUTs, this test case is only mandatory where the NAP supports IPv4 and has no operable routable IPv4 address available (see Section 1.9 in [4]).

- Reference

  [4]

- Test Case Configuration

Test Case
PAN/NAP/IPv4/Autonet/BV-01-C [IP Address Assignment – Ad Hoc Networking (Autonet)]
PAN/GN/IPv4/Autonet/BV-01-C [IP Address Assignment – Ad Hoc Networking (Autonet)]
PAN/PANU/IPv4/Autonet/BV-01-C [IP Address Assignment – Ad Hoc Networking (Autonet)]

Table 4.94: IP Address Assignment – Ad Hoc Networking (Autonet) test cases

- Test Procedure

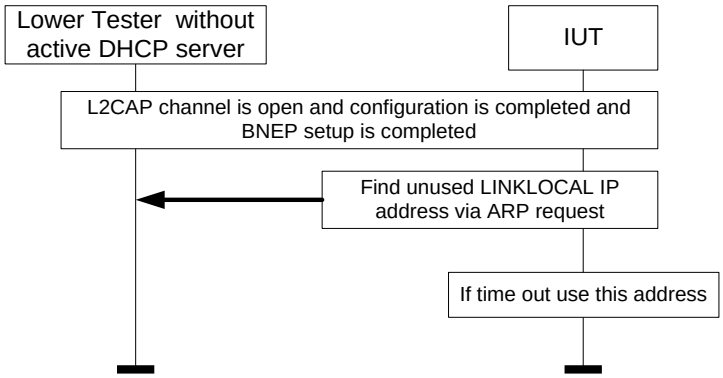

Figure 4.18: IP Address Assignment – Ad Hoc Networking (Autonet)

The IUT chooses a valid LINKLOCAL IP address; for example, an IP address within the range 169.254/16 excluding the first and last 256.

The IUT must check if the selected IP address is already in use by means of issuing an ARP probe. If the selected IP address is already in use (or it receives another ARP probe with the same IP address in), it must select another IP address.

The IUT must try finding a valid IP address until it either finds one or it has tried more than autoconfig-retry count.

- Expected Outcome

  Pass verdict

  The IUT selects an IP address.

4.8.2 IP Address Assignment – Ad Hoc Networking (Autonet)

- Test Purpose

  Verify that an IUT can obtain a valid IPv6 address in an ad hoc networking topology, as described in [6].

- Reference

  [6]

- Test Case Configuration

Test Case
PAN/NAP/IPv6/Autonet/BV-02-C [IP Address Assignment – Ad Hoc Networking (Autonet)]
PAN/GN/IPv6/Autonet/BV-02-C [IP Address Assignment – Ad Hoc Networking (Autonet)]
PAN/PANU/IPv6/Autonet/BV-02-C [IP Address Assignment – Ad Hoc Networking (Autonet)]

Table 4.95: IP Address Assignment – Ad Hoc Networking (Autonet) test cases

- Test Procedure

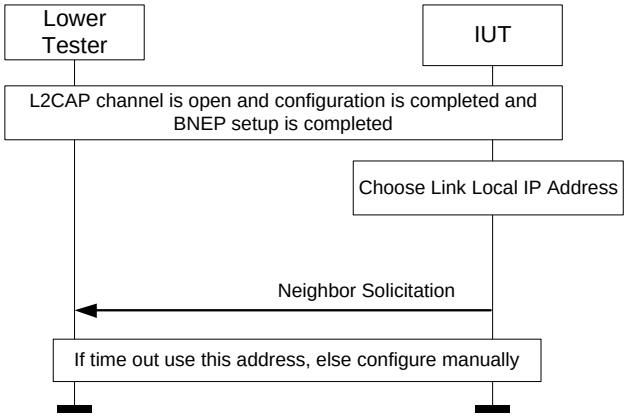

Figure 4.19: IP Address Assignment – Ad Hoc Networking (Autonet)

The IUT chooses a valid LINKLOCAL IP address; that is, an IP address constructed by the LINKLOCAL prefix FE80::0 and the interface identifier.

The IUT checks if the selected IP address is already in use by means of issuing a neighbor solicitation. If auto configuring fails, the IP address is manually configured.

- Expected Outcome

  Pass verdict

  The IUT selects an IP address.

PAN/PANU/IP/DHCP/BV-03-C [IP Address Assignment – Infrastructure (DHCP)]

- Test Purpose

  Verify that an IUT can obtain a valid IP address in an IP infrastructure network topology, as described in [7] and [8].

- Reference

  [7] and [8]

- Test Procedure

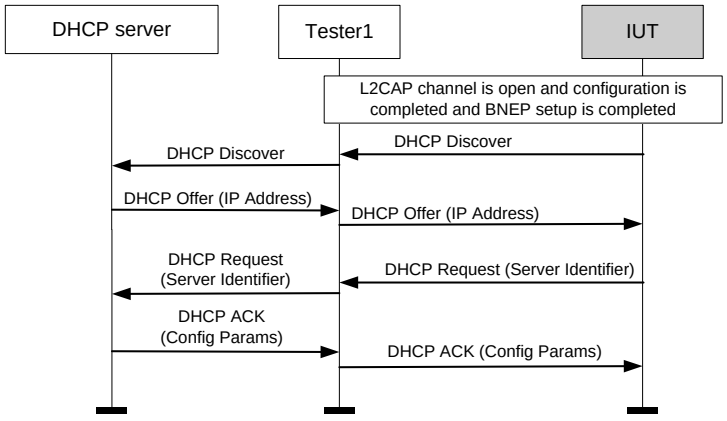

Figure 4.20: PAN/PANU/IP/DHCP/BV-03-C [IP Address Assignment – Infrastructure (DHCP)]

The IUT broadcasts a DHCPDISCOVER packet onto its local subnet.

The DHCP server transmits a DHCPOFFER packet containing an available IP address as well as additional configuration parameters to the IUT.

The IUT responds by broadcasting a DHCPREQUEST with an identifier for the selected DHCP server (to handle the case where multiple DHCP servers responded on the DHCPDISCOVER).

On receipt of the DHCPREQUEST, the selected DHCP server responds with a DHCPACK with the configuration parameters for the IUT.

- Test Condition

  DHCPOFFER, DHCPREQUEST, and DHCPACK packets must conform to their respective specifications. Tester1 must be able to route/bridge traffic between the DHCP server and the IUT. The IUT and Tester1 must be capable of maintaining an open L2CAP connection.

  Note: The two functions (Tester1 and DHCP Server) may not be on physically separate devices. If they are located in different devices, network connectivity must be ensured between these two devices.

- Expected Outcome

  Pass verdict

  The IUT obtains a valid IP address assigned by the DHCP server.

4.8.3 TX LLMNR – Ad Hoc Networking

- Test Purpose

  Verify that an IUT supports LLMNR queries.

- Reference

  PAN Profile [5]

- Test Case Configuration

Test Case
PAN/NAP/IP/LLMNR/BV-01-C [TX LLMNR – Ad Hoc Networking]
PAN/GN/IP/LLMNR/BV-01-C [TX LLMNR – Ad Hoc Networking]
PAN/PANU/IP/LLMNR/BV-01-C [TX LLMNR – Ad Hoc Networking]

Table 4.96: TX LLMNR – Ad Hoc Networking test cases

- Test Procedure

  The IUT requests a name to IP address resolution using LLMNR. Tester1 forwards the request to Tester2. Tester2 responds to the LLMNR query and sends a reply back to Tester1. Tester1 forwards the response from Tester2 to the IUT. Note: The two functions (Tester1 and Tester2) may or may not be on physically separate devices. If they are located in different devices, connectivity must be ensured between these two devices.

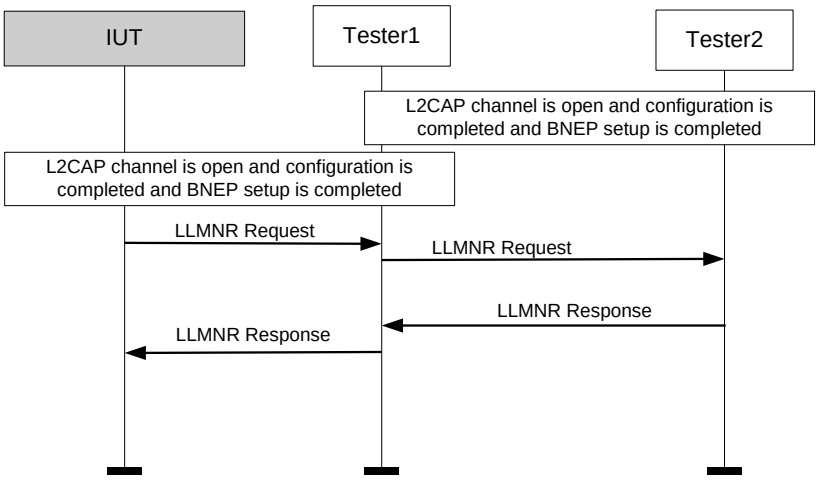

Figure 4.21: TX LLMNR – Ad Hoc Networking

- Test Condition

  The IUT and Tester2 must support LLMNR. the IUT, Tester2, and Tester1 must be capable of maintaining an open L2CAP connection.

- Expected Outcome

  Pass verdict

  The IUT sends a valid LLMNR query to Tester2.

- Notes

  The test is performed individually for each of the protocols (IPv4 and IPv6) supported by the IUT.

4.8.4 LLMNR – Ad Hoc Networking

- Test Purpose

  Verify that an IUT can do LLMNR resolution.

- Reference

  PAN Profile [5]

- Test Case Configuration

Test Case
PAN/NAP/IP/LLMNR/BV-02-C [LLMNR – Ad Hoc Networking]
PAN/GN/IP/LLMNR/BV-02-C [LLMNR – Ad Hoc Networking]
PAN/PANU/IP/LLMNR/BV-02-C [LLMNR – Ad Hoc Networking]

Table 4.97: LLMNR – Ad Hoc Networking test cases

- Test Procedure

  Tester1 requests a name to IP address resolution using LLMNR. Tester2 forwards the request to the IUT. The IUT responds to the LLMNR query and sends a reply back to Tester1. Tester2 forwards the response from the IUT to Tester1. Note: The two functions (Tester1 and Tester2) may or may not be on physically separate devices. If they are located in different devices, connectivity must be ensured between these two devices.

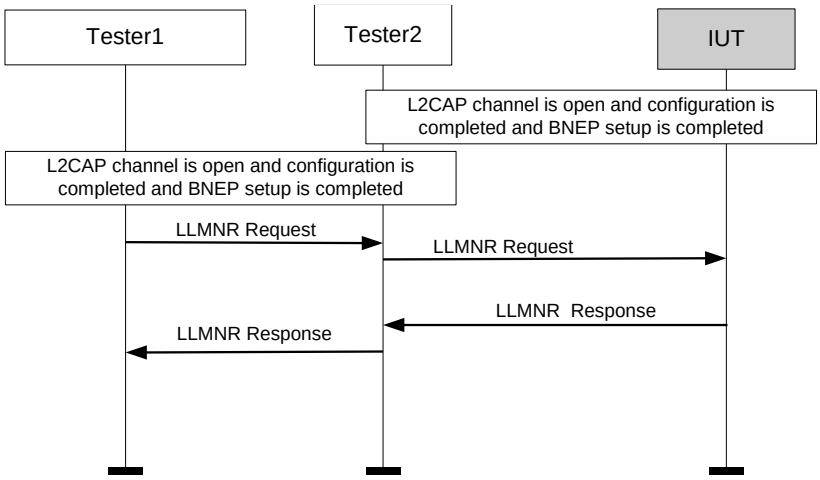

Figure 4.22: LLMNR – Ad Hoc Networking

- Test Condition

  Tester1 and the IUT must support LLMNR. Tester1, the IUT, and Tester2 must be capable of maintaining an open L2CAP connection.

- Expected Outcome

  Pass verdict

  Tester1 (PANU) receives a valid DNS reply from the IUT (PANU).

- Notes

  The test is performed individually for each of the protocols (IPv4 and IPv6) supported by the IUT.

4.8.5 DNS Resolution – infrastructure

- Test Purpose

  Verify that a PANU can do DNS resolution through a NAP in an infrastructure topology.

- Reference

  [9] and [10]

- Test Case Configuration

Test Case
PAN/NAP/IP/DNS/BV-01-C [DNS Resolution – infrastructure]
PAN/PANU/IP/DNS/BV-01-C [DNS Resolution – infrastructure]

Table 4.98: DNS Resolution – infrastructure test cases

- Test Procedure

  The PANU requests a name to IP address resolution using a domain name service on Tester2 (DNS Server). The NAP forwards the PANU request to the DNS Server. The DNS Server responds to the DNS query by sending a DNS reply to the PANU. The NAP forwards the response from the DNS Server to the PANU.

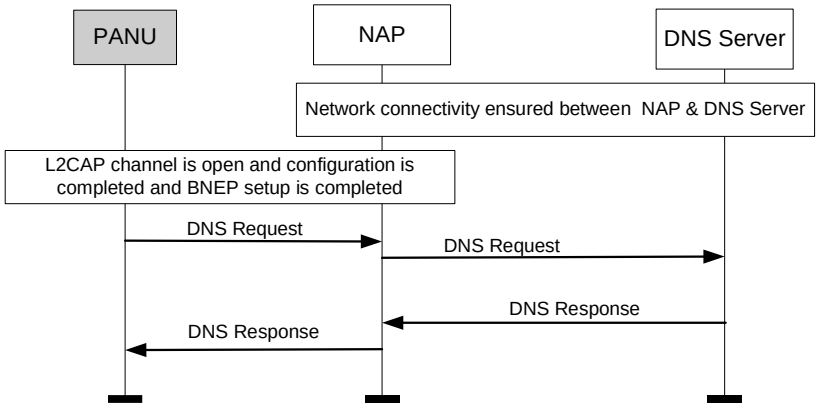

Figure 4.23: DNS Resolution – infrastructure

- Test Condition

  The NAP must be able to bridge/route IP traffic between the PANU and the DNS Server. The PANU and the NAP must be capable of maintaining an open L2CAP connection. Note: The two functions (NAP and DNS Server) may or may not be on physically separate devices. If they are located in different devices, network connectivity must be ensured between these two devices.

- Expected Outcome

  Pass verdict

  The PANU receives a valid DNS response from the DNS Server.

- Notes

  The test is performed individually for each of the protocols (IPv4 and IPv6) supported by the IUT.

4.8.6 Basic Data Transmission (Look up WEB Page)

- Test Purpose

  Verify the capability of a basic data transmission by looking up a web page. A device must be able to pass this test case if the device supports HTTP.

- Reference

  PAN Profile [5]

- Test Case Configuration

Test Case
PAN/NAP/IP/APP/BV-01-C [Basic Data Transmission (Look up WEB Page)]
PAN/GN/IP/APP/BV-01-C [Basic Data Transmission (Look up WEB Page)]
PAN/PANU/IP/APP/BV-01-C [Basic Data Transmission (Look up WEB Page)]

Table 4.99: Basic Data Transmission (Look up WEB Page) test cases

- Test Procedure

  The IUT requests a web page from the web server running in the Lower Tester in terms of issuing an HTTP request. The IUT receives an HTTP reply containing the data specified by the URL in the HTTP request.

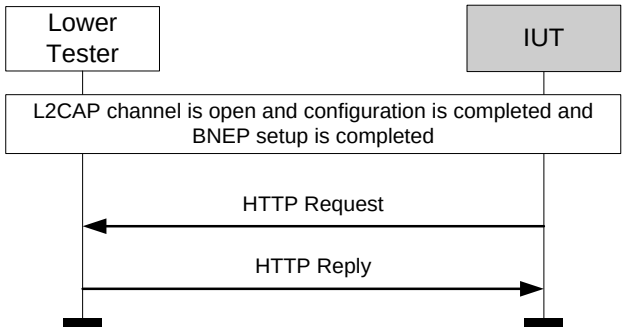

Figure 4.24: Basic Data Transmission (Look up WEB Page)

- Test Condition

  The Lower Tester is running a web server. The Lower Tester and the IUT must be capable of maintaining an open L2CAP connection.

- Expected Outcome

  Pass verdict

  The IUT receives the data requested by the HTTP request and delivers it to the requesting application.

4.8.7 Basic Data Transmission (Look Up WAP Page)

- Test Purpose

  Verify the capability of a basic data transmission by looking up a WAP page. A device must be able to pass this test case if the device supports WAP.

- Reference

  PAN Profile [5]

- Test Case Configuration

Test Case
PAN/NAP/IP/APP/BV-02-C [Basic Data Transmission (Look Up WAP Page)]
PAN/GN/IP/APP/BV-02-C [Basic Data Transmission (Look Up WAP Page)]
PAN/PANU/IP/APP/BV-02-C [Basic Data Transmission (Look Up WAP Page)]

Table 4.100: Basic Data Transmission (Look Up WAP Page) test cases

- Test Procedure

  The IUT requests a WAP page from the WAP server running in the Lower Tester in terms of issuing a WAP request. The IUT receives a WAP reply containing the data specified by the URL in the WAP request.

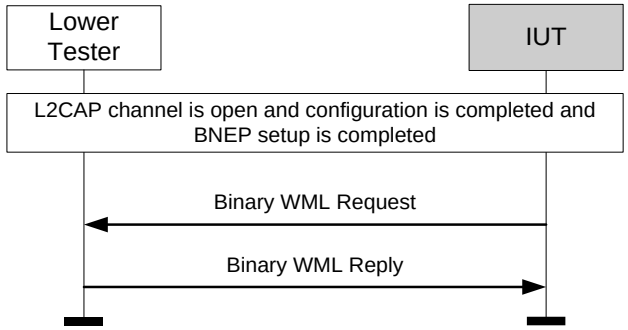

Figure 4.25: Basic Data Transmission (Look Up WAP Page)

- Test Condition

  The Lower Tester and the IUT must be within radio range of each other and have both validly configured an IP address. The Lower Tester is running a WAP server. The Lower Tester and the IUT must be capable of maintaining an open L2CAP connection.

- Expected Outcome

  Pass verdict

  The IUT receives the data requested by the WAP request and delivers it to the requesting application.

4.8.8 Control Messages (IUT Sends Ping)

- Test Purpose

  Verify the capability of sending and receiving control messages. A device must be able to pass this test case if the device supports IP.

- Reference

  [11] and [12]

- Test Case Configuration

Test Case
PAN/NAP/IP/APP/BV-03-C [Control Messages (IUT Sends Ping)]
PAN/GN/IP/APP/BV-03-C [Control Messages (IUT Sends Ping)]
PAN/PANU/IP/APP/BV-03-C [Control Messages (IUT Sends Ping)]

Table 4.101: Control Messages (IUT Sends Ping) test cases

- Test Procedure

  The IUT sends an ICMP echo request to the Lower Tester using the proper IP address.

  The Lower Tester responds to the request by sending an ICMP echo reply to the IUT.

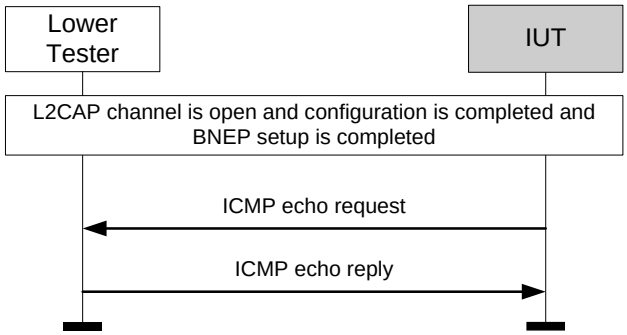

Figure 4.26: Control Messages (IUT Sends Ping)

- Test Condition

  The Lower Tester and the IUT both have valid IP addresses configured.

- Expected Outcome

  Pass verdict

  The IUT receives the ICMP echo reply.

PAN/PANU/IP/APP/BV-04-C [Periodically Checking for DHCP Service]

- Test Purpose

  Verify the capability of detecting and using DHCP service after configuration of a link-local IP address. A device must be able to pass this test case if the device supports DHCP.

- Reference

  [4] and [7]

- Test Procedure

  Perform stateless link-local IP address assignment by [4] or [7].

  A DHCP server becomes available on the link.

  The IUT must configure a new IP address. If the IUT is able to handle multiple IP addresses, any active IP connections may be maintained. If the IUT is able to handle one single IP address at the time, any active applications are terminated.

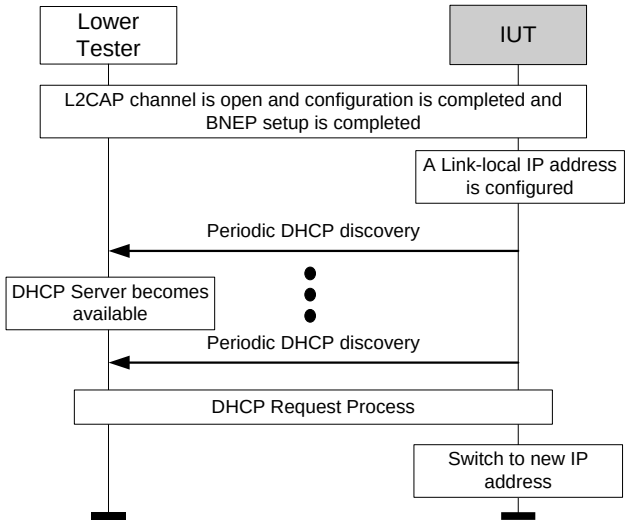

Figure 4.27: PAN/PANU/IP/APP/BV-04-C [Periodically Checking for DHCP Service]

- Test Condition

  The Lower Tester and the IUT have both configured a valid IP address through a link-local IP address autoconfiguration mechanism. The Lower Tester must be able to provide DHCP service, after the Lower Tester and the IUT have set up their BNEP connection.

- Expected Outcome

  Pass verdict

  The IUT configures the IP address assigned by the DHCP server.

  The IUT periodically tries to obtain a valid IP address assigned from a DHCP server.

4.8.9 Control Messages (Device Responds to Ping)

- Test Purpose

  Verify the ability to send and receive IP Ping (ICMP echo request) control messages.

- Reference

  [11] and [12]

- Test Case Configuration

Test Case
PAN/NAP/IP/APP/BV-05-C [Control Messages (Device Responds to Ping)]
PAN/GN/IP/APP/BV-05-C [Control Messages (Device Responds to Ping)]
PAN/PANU/IP/APP/BV-05-C [Control Messages (Device Responds to Ping)]

Table 4.102: Control Messages (Device Responds to Ping) test cases

- Test Procedure

  The Lower Tester sends an ICMP echo request to the IUT using the proper IP address.

  The IUT responds to the request by sending an ICMP echo reply to the Lower Tester.

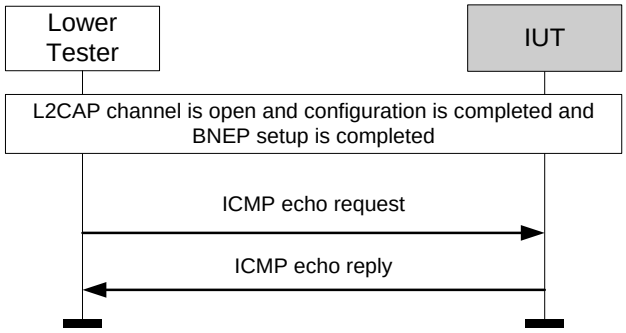

Figure 4.28: Control Messages (Device Responds to Ping)

- Test Condition

  The Lower Tester and the IUT have both configured valid IP addresses. The Lower Tester and the IUT must be capable of maintaining an open L2CAP connection.

- Expected Outcome

  Pass verdict

  The IUT sends the ICMP echo reply to the Lower Tester.

4.9 PAN profile Miscellaneous Conformance test cases

This section describes various miscellaneous Conformance test cases for the PAN profile.

PAN/PANU/MISC/ROLE/BV-01-C [PANU Performs Role Switch after LM Connection Establishment]

- Test Purpose

  Verify that the IUT will accept and perform a baseband role switch to the peripheral role when requested by the Lower Tester after LM connection establishment.

- Reference

  [5] 11.1

- Initial Condition

  - The IUT is in the PANU role. The Lower Tester is a multi-user NAP/GN.
  - The IUT initiates the baseband page procedure and connects to the Lower Tester, which does not request a role switch during LM connection establishment.

- Test Procedure

  1. The IUT requests the opening of an L2CAP channel for BNEP.
  2. The Lower Tester requests a role switch.
  3. The IUT accepts this request and becomes a peripheral of the piconet.

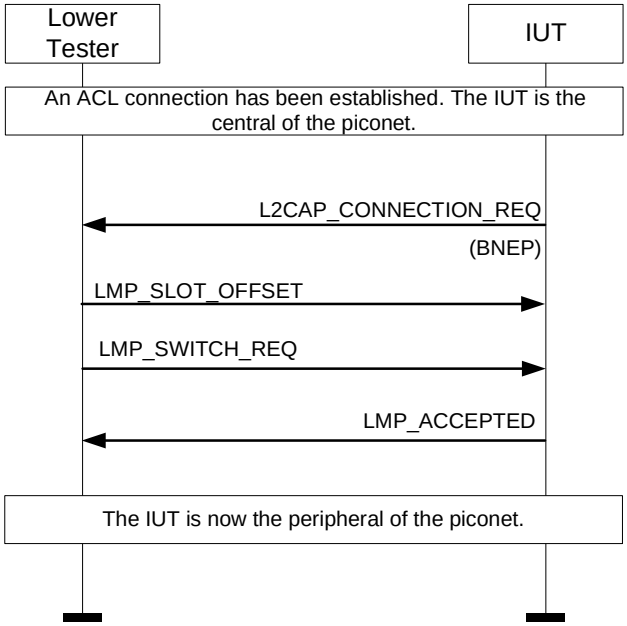

Figure 4.29: PAN/PANU/MISC/ROLE/BV-01-C [PANU Performs Role Switch after LM Connection Establishment]

- Expected Outcome

  Pass verdict

  The IUT successfully accepts the role switch requested by the Lower Tester and becomes a peripheral of the piconet.

- Notes

  Not all LMP messages exchanged are shown.

PAN/PANU/MISC/ROLE/BV-02-C [PANU Performs Role Switch during LM Connection Establishment]

- Test Purpose

  Verify that the IUT will accept and perform a baseband role switch to the peripheral role when requested by the Lower Tester during LM connection establishment.

- Reference

  [5] 11.1

- Initial Condition

  - The IUT is in the PANU role. The Lower Tester is a multi-user NAP/GN.

- Test Procedure

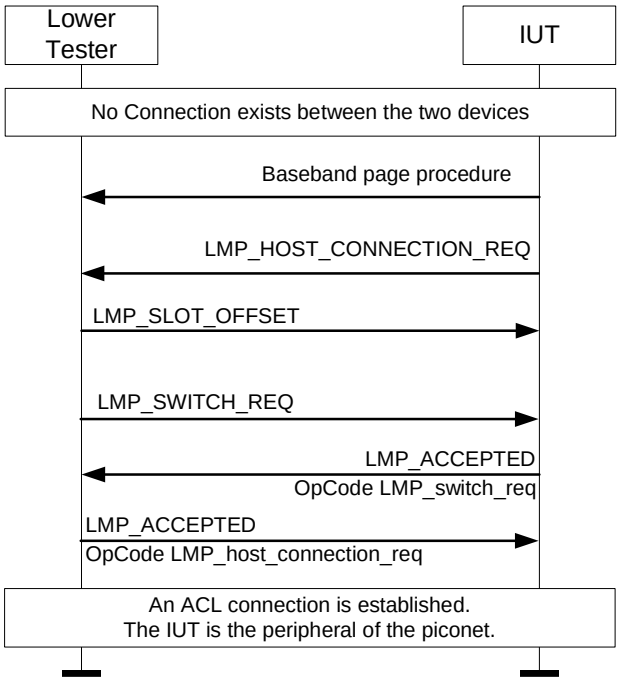

Figure 4.30: PAN/PANU/MISC/ROLE/BV-02-C [PANU Performs Role Switch during LM Connection Establishment]

1. The IUT pages the Lower Tester.
2. The Lower Tester requests a role switch during LM connection establishment.
3. The IUT accepts this request and becomes a peripheral of the piconet.

- Expected Outcome

  Pass verdict

  The IUT successfully accepts the role switch requested by the Lower Tester and becomes the peripheral of the piconet.

- Notes

  Not all LMP messages exchanged are shown.

4.9.1 32-Bit UUID processing

- Test Purpose

  Verify that BNEP connection setup 32-bit UUIDs are processed correctly.

  This test involves two Bluetooth devices, with one of the Bluetooth devices attempting to establish a BNEP connection using 32-bit UUIDs.

- Reference

  BNEP [1]

- Initial Condition

  - An L2CAP channel is open and configuration has completed.

- Test Case Configuration

Test Case
PAN/NAP/MISC/UUID/BV-01-C [32-Bit UUID processing]
PAN/GN/MISC/UUID/BV-01-C [32-Bit UUID processing]
PAN/PANU/MISC/UUID/BV-01-C [32-Bit UUID processing]

Table 4.103: 32-Bit UUID processing test cases

- Test Procedure

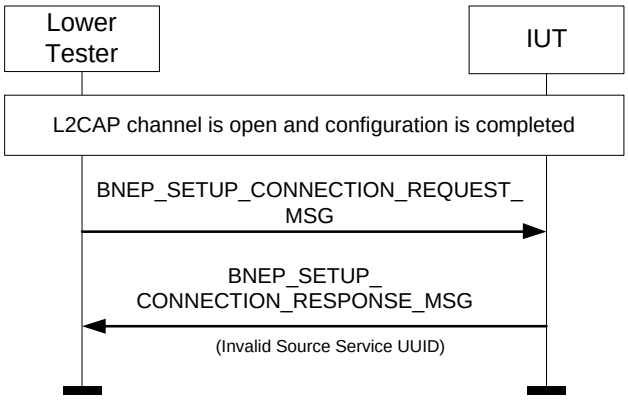

Figure 4.31: 32-Bit UUID processing

The Lower Tester establishes an L2CAP connection for BNEP.

The Lower Tester requests a BNEP connection setup using the following 32-bit UUIDs, depending on the IUT's role:

IUT Role	Destination UUID	Source UUID
PANU	0x00001115	0x0bad1116
NAP	0x00001116	0x0bad1115
GN	0x00001117	0x0bad1115

Table 4.104: 32-Bit UUID processing

The device under test rejects this request with reason Invalid Source Service UUID (0x0002).

- Expected Outcome

  Pass verdict

  The device under test rejects the BNEP connection setup request with reason Invalid Source Service UUID (0x0002).

4.9.2 128-Bit UUID processing

- Test Purpose

  Verify that 128-bit UUIDs are processed correctly.

- Reference

  BNEP [1]

- Test Case Configuration

Test Case
PAN/NAP/MISC/UUID/BV-02-C [128-Bit UUID processing]
PAN/GN/MISC/UUID/BV-02-C [128-Bit UUID processing]
PAN/PANU/MISC/UUID/BV-02-C [128-Bit UUID processing]

Table 4.105: 128-Bit UUID processing test cases

- Test Procedure

  The Lower Tester establishes an L2CAP connection for BNEP.

  The Lower Tester requests a BNEP connection setup using the following 128-bit UUIDs, depending on the IUT's role:

IUT Role	Destination UUID	Source UUID
PANU	0x00001115000010008000000805f9b34fb	0x000011160000100080bad0805f9b34fb
NAP	0x000011160000100080000000805f9b34fb	0x000011150000100080bad0805f9b34fb
GN	0x000011170000100080000000805f9b34fb	0x000011150000100080bad0805f9b34fb

Table 4.106: 128-Bit UUID processing

  The device under test rejects this request with reason Invalid Source Service UUID (0x0002).

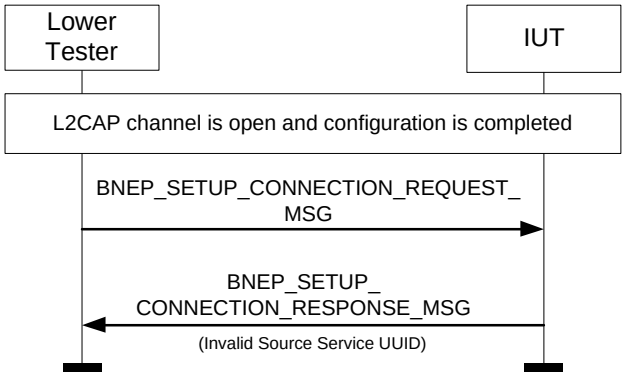

Figure 4.32: 128-Bit UUID processing

- Expected Outcome

  Pass verdict

  The device under test rejects the BNEP connection setup request with reason Invalid Source Service UUID (0x0002).

5 Test case mapping

The Test Case Mapping Table (TCMT) maps test cases to specific requirements in the ICS. The IUT is tested in all roles for which support is declared in the ICS document.

The columns for the TCMT are defined as follows:

Item: Contains a logical expression based on specific entries from the associated ICS document. Contains a logical expression (using the operators AND, OR, NOT as needed) based on specific entries from the applicable ICS document(s). The entries are in the form of y/x references, where y corresponds to the table number and x corresponds to the feature number as defined in the ICS document for the PAN Profile [14].

Feature: A brief, informal description of the feature being tested.

Test Case(s): The applicable test case identifiers are required for Bluetooth Qualification if the corresponding y/x references defined in the Item column are supported. Further details about the function of the TCMT are elaborated in [13].

For the purpose and structure of the ICS/IXIT, refer to [13].

Item	Feature	Test Case(s)
PAN 2/2 AND PAN 2/19	Support BNEP Forwarding	PAN/NAP/BNEP/FORWARD-BROADCAST/BV-09-C
PAN 3/2 AND PAN 3/17	Support BNEP Forwarding	PAN/GN/BNEP/FORWARD-BROADCAST/BV-09-C
PAN 2/2 AND PAN 2/20	Support BNEP Forwarding	PAN/NAP/BNEP/EXTENSION-0/BV-07-C PAN/NAP/BNEP/FORWARD/BV-08-C
PAN 3/2 AND PAN 3/18	Support BNEP Forwarding	PAN/GN/BNEP/EXTENSION-0/BV-07-C PAN/GN/BNEP/FORWARD/BV-08-C
PAN 2/3	Support Layer 2-Bridging between PAN and External Network	PAN/NAP/BNEP/BROADCAST-0/BV-01-C PAN/NAP/BNEP/BROADCAST-1/BV-02-C PAN/NAP/BNEP/MULTICAST-0/BV-03-C PAN/NAP/BNEP/MULTICAST-1/BV-04-C PAN/NAP/BNEP/FORWARD-UNICAST/BV-05-C PAN/NAP/BNEP/FORWARD-UNICAST/BV-06-C
PAN 2/4	Support IP Forwarding between PAN and External Network	PAN/NAP/BRIDGE/TX/BV-01-C PAN/NAP/BRIDGE/RX/BV-02-C
PAN 2/5 AND PAN 2/20	Support BNEP Packet Filtering	PAN/NAP/BNEP/FILTER/BV-10-C PAN/NAP/BNEP/FILTER/BV-11-C PAN/NAP/BNEP/FILTER/BV-12-C PAN/NAP/BNEP/FILTER/BV-13-C PAN/NAP/BNEP/FILTER/BV-14-C PAN/NAP/BNEP/FILTER/BV-15-C
PAN 3/3 AND PAN 3/18	Support BNEP Packet Filtering	PAN/GN/BNEP/FILTER/BV-10-C PAN/GN/BNEP/FILTER/BV-11-C PAN/GN/BNEP/FILTER/BV-12-C PAN/GN/BNEP/FILTER/BV-13-C PAN/GN/BNEP/FILTER/BV-14-C PAN/GN/BNEP/FILTER/BV-15-C

Item	Feature	Test Case(s)
PAN 2/6b	Support IPv4 link-local address configuration	PAN/NAP/IPv4/Autonet/BV-01-C
PAN 3/4	Support IPv4 link-local address configuration	PAN/GN/IPv4/Autonet/BV-01-C
PAN 4/2	Support IPv4 link-local address configuration	PAN/PANU/IPv4/Autonet/BV-01-C
PAN 1/3 AND PAN 4/4	Support DHCP Client for IPv4	PAN/PANU/IP/DHCP/BV-03-C PAN/PANU/IP/APP/BV-04-C
PAN 2/9 OR PAN 2/14	Support DNS Requester	PAN/NAP/IP/DNS/BV-01-C
PAN 4/5 OR PAN 4/10	Support DNS / Resolver	PAN/PANU/IP/DNS/BV-01-C
PAN 2/9a OR PAN 2/14a	Support LLMNS Sender	PAN/NAP/IP/LLMNR/BV-01-C
PAN 3/7a OR PAN 3/12a	Support LLMNS Sender	PAN/GN/IP/LLMNR/BV-01-C
PAN 4/5a OR PAN 4/10a	Support LLMNS Sender	PAN/PANU/IP/LLMNR/BV-01-C
PAN 2/9b OR PAN 2/14b	Support LLMNS Responder	PAN/NAP/IP/LLMNR/BV-02-C
PAN 3/7b OR PAN 3/12b	Support LLMNS Responder	PAN/GN/IP/LLMNR/BV-02-C
PAN 4/5b OR PAN 4/10b	Support LLMNS Responder	PAN/PANU/IP/LLMNR/BV-02-C
PAN 2/10 OR PAN 2/15	Support HTTP Client for IPv4, and IPv6	PAN/NAP/IP/APP/BV-01-C
PAN 3/8 OR PAN 3/13	Support HTTP Client for IPv4, and IPv6	PAN/GN/IP/APP/BV-01-C
PAN 4/6 OR PAN 4/11	Support HTTP Client for IPv4, and IPv6	PAN/PANU/IP/APP/BV-01-C
PAN 2/16 OR PAN 2/11	Support WAP Client for IPv4 and IPv6	PAN/NAP/IP/APP/BV-02-C
PAN 3/14 OR PAN 3/9	Support WAP Client for IPv4 and IPv6	PAN/GN/IP/APP/BV-02-C
PAN 4/12 OR PAN 4/7	Support WAP Client for IPv4 and IPv6	PAN/PANU/IP/APP/BV-02-C
PAN 2/12	Support IPv6	PAN/NAP/IPv6/Autonet/BV-02-C
PAN 3/10	Support IPv6	PAN/GN/IPv6/Autonet/BV-02-C
PAN 4/8	Support IPv6	PAN/PANU/IPv6/Autonet/BV-02-C
PAN 2/6 OR PAN 2/12	Support IPv4 and IPv6	PAN/NAP/IP/APP/BV-05-C
PAN 3/4 OR PAN 3/10	Support IPv4 and IPv6	PAN/GN/IP/APP/BV-05-C
PAN 4/2 OR PAN 4/8	Support IPv4 and IPv6	PAN/PANU/IP/APP/BV-05-C
PAN 2/7 OR PAN 2/13	Support Ping Client for IPV4 and IPv6	PAN/NAP/IP/APP/BV-03-C

Item	Feature	Test Case(s)
PAN 3/5 OR PAN 3/11	Support Ping Client for IPV4 and IPv6	PAN/GN/IP/APP/BV-03-C
PAN 4/3 OR PAN 4/9	Support Ping Client for IPV4 and IPv6	PAN/PANU/IP/APP/BV-03-C
PAN 4/13	Support connections to multi-user NAPs/GNs	PAN/PANU/MISC/ROLE/BV-01-C PAN/PANU/MISC/ROLE/BV-02-C
PAN 2/17	Supports Connectable Mode	PAN/NAP/MISC/UUID/BV-01-C PAN/NAP/MISC/UUID/BV-02-C
PAN 3/15	Supports Connectable Mode	PAN/GN/MISC/UUID/BV-01-C PAN/GN/MISC/UUID/BV-02-C
PAN 4/14	Supports Connectable Mode	PAN/PANU/MISC/UUID/BV-01-C PAN/PANU/MISC/UUID/BV-02-C
PAN 2/18	PAN NAP SDP Service	PAN/NAP/SGSIT/SERR/BV-01-C PAN/NAP/SGSIT/ATTR/BV-01-C PAN/NAP/SGSIT/ATTR/BV-02-C PAN/NAP/SGSIT/ATTR/BV-03-C PAN/NAP/SGSIT/ATTR/BV-04-C PAN/NAP/SGSIT/ATTR/BV-05-C PAN/NAP/SGSIT/ATTR/BV-06-C PAN/NAP/SGSIT/ATTR/BV-07-C PAN/NAP/SGSIT/ATTR/BV-08-C PAN/NAP/SGSIT/ATTR/BV-09-C PAN/NAP/SGSIT/OFFS/BV-01-C PAN/NAP/SGSIT/OFFS/BV-02-C
PAN 3/16	PAN GN SDP Service	PAN/GN/SGSIT/SERR/BV-02-C PAN/GN/SGSIT/ATTR/BV-10-C PAN/GN/SGSIT/ATTR/BV-11-C PAN/GN/SGSIT/ATTR/BV-12-C PAN/GN/SGSIT/ATTR/BV-13-C PAN/GN/SGSIT/ATTR/BV-14-C PAN/GN/SGSIT/ATTR/BV-15-C PAN/GN/SGSIT/ATTR/BV-16-C PAN/GN/SGSIT/OFFS/BV-03-C PAN/GN/SGSIT/OFFS/BV-04-C
PAN 4/15	PAN PANU SDP Service	PAN/PANU/SGSIT/SERR/BV-03-C PAN/PANU/SGSIT/ATTR/BV-17-C PAN/PANU/SGSIT/ATTR/BV-18-C PAN/PANU/SGSIT/ATTR/BV-19-C PAN/PANU/SGSIT/ATTR/BV-20-C PAN/PANU/SGSIT/ATTR/BV-21-C PAN/PANU/SGSIT/OFFS/BV-05-C PAN/PANU/SGSIT/OFFS/BV-06-C
PAN 1/3	Successful Connection with future SDP Record value – PAN PANU	PAN/PANU/CGSIT/SFC/BV-01-C

Table 5.1: Test case mapping

6 Revision history and acknowledgments

Revision History

Publication Number	Revision Number	Date	Comments
0	1.0	2003-02-14	Specification Adopted
	1.0.1r1	2005-02-11	Editorial and format changes. Change document numbering. Incorporate TSE 618 for TCMT. Incorporate TSE 624 for TP/PAN/BNEP/EXTENSION-07/BV-07-C.
1	1.0.1	2005-02-18	Prepare for publication.
	1.0.2r1	2005-08-31	TSE 823: TP/PAN/BNEP/EXTENSION-0/BV-07-C, TP/PAN/BNEP/FORWARD/BV-08-C, TP/PAN/BNEP/FORWARD-BROADCAST/BV-09-C, TP/PAN/BNEP/FILTER*
	1.0.2r2	2005-10-12	Correct selection expression in TCMT
2	1.0.2	2005-10-21	Prepare for publication.
	1.0.3r0-2	2006-11-01 – 2006-12-01	TSE 1796: TP/PAN/IPv4/Autonet/BV-01-I 1) add new TCMT row, 2) change test purpose TSE 1784:Change TCMT and Notes § for TP/PAN/IP/LLMNR/BV-01-I, TP/PAN/IP/LLMNR/BV-02-I, TP/PAN/IP/DNS/BV-01-I Moved TOC to below Contributors table Input reviewer's comments Changed adopted spec version to 1.1 Corrected previous Comments entry from TSE 1894 to TSE 1784 Input changes from reviewer's copy: Redundant abbreviation section removed Uncertainties removed. TSE 1996. TP/PAN/BNEP/BROADCAST-0/BV-01-C, TP/PAN/BNEP/BROADCAST-1/BV-02-C, TP/PAN/BNEP/MULTICAST-0/BV-03-C, TP/PAN/BNEP/MULTICAST-1/BV-04-C, TP/PAN/BNEP/FORWARD-UNICAST/BV-05-C, TP/PAN/BNEP/FORWARD-UNICAST/BV-06-C Removed unreferenced listings from Reference Table. Inserted sentence in first paragraph. Inserted cross references in Reference section. Spell checked.
3	1.0.3	2007-01-08	Prepare for publication.
	1.0.4r0-1	2008-09-18 - 2008-10-18	TSE 2441:TP/PAN/BRIDGE/TX/BV-01-I: Remove sentence TSE 2634: Update test purposes for TP/PAN/IP/APP/BV-01-I, TP/PAN/IP/APP/BV-02-I, TP/PAN/IP/APP/BV-03-I, TP/PAN/IP/APP/BV-05-I TSE 2647: TP/PAN/IP/APP/BV-03-I: change title
4	1.0.4	2008-12-05	Prepare for publication.

Publication Number	Revision Number	Date	Comments
5	1.0.5	2009-08-10	TSE 2951: TCMT corrections to consolidate entries for IPv4 and IPv6
	1.0.6r0	2011-11-10	TSE 4140: TP/PAN/IP/DNS/BV-01-I: Change diagram and all references in test case to PANU, NAP, DNS-Server TSE 3325: TP/PAN/IP/DHCP/BV-03-I, TP/PAN/IP/APP/BV-04-C; change TCMT
6	1.0.6	2012-03-30	Prepare for publication.
	1.0.7r00	2014-10-21	TSE 5980: Added additional naming conventions to the TP Naming Conventions section.
7	1.0.7	2014-12-08	Prepare for TCRL 2014-2 publication
	1.0.8r00	2015-01-21	Template Conversion Updated Tester1 to Lower Tester where appropriate. TSE 5663: Update testing topologies section to reflect current profile use cases and remove dependencies on testers with wired Ethernet: TP/PAN/BNEP/BROADCAST-0/BV-01-C TP/PAN/BNEP/BROADCAST-1/BV-02-C TP/PAN/BNEP/MULTICAST-0/BV-03-C TP/PAN/BNEP/MULTICAST-1/BV-04-C
8	1.0.8	2015-07-14	Prepared for TCRL 2015-1 publication
	1.0.9r00	2016-08-31	Converted to new Test Case ID conventions as defined in TSTO v4.1.
	1.0.9r01	2016-11-15	Fixed header styles and regenerated ToC
	1.0.9r02	2016-11-22	Fixed TCMT per new TC ID renaming
9	1.0.9	2016-12-13	Approved by BTI. Prepared for TCRL 2016-2 publication.
	1.0.10r00-r01	2018-10-03 – 2018-11-04	TSE 10899 (rating 4): Updated test purpose and deleted test case ID PAN/GN/IP/DNS/BV-01-I for PAN DNS Test Case 1 DNS Resolution – infrastructure. In the TCMT, updated feature for test case PAN/NAP/IP/DNS/BV-01-I and deleted test case PAN/GN/IP/DNS/BV-01-I. Updated test suite template.
10	1.0.10	2018-11-21	Approved by BTI. Prepared for TCRL 2018-2 publication.
	p11r00–r04	2021-10-18 – 2021-12-07	TSE 17200 (rating 1): Editorial rewrite of the PAN TS to update the conventions used in other Test Suites. Incorporated consistency checker editorials. Performed template-related fixes. Updated the introduction text before the TCMT to align with the template. Reverted “central” to “master” and “peripheral” to “slave” throughout for consistency with the spec. Updated copyright page to align with v2 of the DNMD.
11	p11	2022-01-25	Approved by BTI on 2021-12-19. Prepared for TCRL 2021-2 publication.

Publication Number	Revision Number	Date	Comments
	p12r00–r03	2023-11-07 – 2024-04-08	TSE 23924 (rating 1): Converted -I tests to -C tests as appropriate; updated the TCMT and TCRL accordingly. Renamed Sections 4.6 and 4.7 per BTI recommendation to remove “interoperability” association. TSE 24533 (rating 4): Added new GSIT section with new TCs PAN/NAP/SGSIT/SERR/BV-01-C, PAN/NAP/SGSIT/OFFS/BV-01-C and -02-C, PAN/NAP/SGSIT/ATTR/BV-01-C – -09-C, PAN/GN/SGSIT/SERR/BV-02-C, PAN/GN/SGSIT/OFFS/BV-03-C and -04-C, PAN/GN/SGSIT/ATTR/BV-10-C – -16-C, PAN/PANU/CGSIT/SFC/BV-01-C, PAN/PANU/SGSIT/SERR/BV-03-C, PAN/PANU/SGSIT/OFFS/BV-05-C and -06-C, and PAN/PANU/SGSIT/ATTR/BV-17-C – -21-C. Deleted “PAN profile SDP Conformance test cases” including TCs PAN/NAP/SDP/BV-01-C, PAN/GN/SDP/BV-02-C, and PAN/PANU/SDP/BV-03-C. Updated the TCMT accordingly. Added references to the SDP TS, the PAN IXIT, and the Appropriate Language document. Updated the Test Groups and TC Conventions sections. Modernized test language globally, including updating “master” to “central” and “slave” to “peripheral”. Template-related editorials, including revising “shall” statements, moving TCIDs to TC Config tables, and aligning boilerplate text with the latest TS template, as appropriate.
12	p12	2024-07-01	Approved by BTI on 2024-05-22. Prepared for TCRL 2024-1 publication.

Acknowledgments

Name	Company
David Moore	3COM Corporation
Tom Scribner	3COM Corporation
Barry Corlett	Agere Systems
Willy Sagefalk	Axis Communications
Alicia Courtney	Broadcom Corporation
Dan Willey	Certicom Corporation
Horia Balog	Classwave Wireless Inc.
Conrad Maxwell	Conexant Systems
Mark Rison	CSR
Magnus Sommansson	CSR
Allan Bogeskov	Ericsson
Theo Borst	Ericsson
Per Johansson	Ericsson
Tero Kauppinen	Ericsson

Name	Company
Martin Kitchen	Ericsson
Jesper Krogh	Ericsson
Tony Larsson	Ericsson
Johan Sorensen	Ericsson
Dave Suvak	Extended Systems Inc.
Jean Tourrilhes	Hewlett Packard Corporation
Toru Aihara	International Business Machines Corporation
Chatschik Bisdikian	International Business Machines Corporation
Kris Fleming	Intel Corporation
Robert Hunter	Intel Corporation
Jon Inouye	Intel Corporation
Eiji Kato	Matsushita Electric Industrial
Billy Brackenridge	Microsoft Corporation
Mike Foley	Microsoft Corporation
Dale Farnsworth	Motorola Inc.
Brian Redding	Motorola Inc.
Carmen Kuhl	Nokia Corporation
Jaakko Lipasti	Nokia Corporation
James Scales	Nokia Corporation
Markus Schetelig	Nokia Corporation
Sander van Valkenburg	Nokia Corporation
Steven Kenny	Norwood Systems
Rebecca Ostergaard	Norwood Systems
Graeme Reid	Norwood Systems
Diego Melpignano	Philips Inc.
Darrell Goff	Rappore
Simon Harrison	Red-M Communications Ltd
Daniel Shaw	Red-M Communications Ltd
Pravin Bhagwat	ReefEdge, Inc.
Daryl Hlasny	Sharp Laboratories of America Inc.
Leonard Ott	Socket Communications Inc.
Wilhelm Hagg	Sony Corporation
Johannes Loebbert	Sony Corporation
Takashi Sasai	Sony Corporation
Mike Blackstock	Synchropoint Wireless, Inc.
Tatuya Jinmei	Toshiba Corporation
Kazuo Nogami	Toshiba Corporation
Yosuke Tajika	Toshiba Corporation
Jim Hobza	Widcomm Inc.
Ravindranath Singamneni	Widcomm Inc.