ETSI TS 104 153-1 V1.1.1 (2026-02)

TECHNICAL SPECIFICATION

Core Network and Interoperability Testing (INT); Conformance Testing for IP/ICMP Translation Algorithm; (IETF RFC 7915); Part 1: Protocol Implementation Conformance Statement (PICS)

Reference
DTS/INT-00208-1

Keywords
conformance, IPv6, PICS

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI IPR online database.

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™**, **LTE™** and **5G™** logo are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Core Network and Interoperability Testing (INT).

The present document is part 1 of a multi-part deliverable covering the test specifications for the IP/ICMP Translation Algorithm, as identified below:

Part 1: "Protocol Implementation Conformance Statement (PICS)";

Part 2: "Test Suite Structure (TSS) and Test Purposes (TP)";

Part 3: "Abstract Test Suite (ATS) and partial Protocol Implementation eXtra Information for Testing (PIXIT) pro forma specification".

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Introduction

To evaluate protocol conformance of a particular implementation, it is necessary to have a statement of which capabilities and options have been implemented for a telecommunication specification. Such a statement is called a Protocol Implementation Conformance Statement (PICS).

1 Scope

The present document provides the Protocol Implementation Conformance Statement (PICS) pro forma for the test specification for the IP/ICMP Translation Algorithm as specified in IETF RFC 7915 [1] in compliance with the relevant requirements and in accordance with the relevant guidance given in ISO/IEC 9646-7 [2] and ETSI ETS 300 406 [i.2].

The supplier of a protocol implementation which is claimed to conform to IETF RFC 7915 [1] is required to complete a copy of the PICS pro forma provided in annex A of the present document and is required to provide the information necessary to identify both the supplier and the implementation.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found in the ETSI docbox.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents are necessary for the application of the present document.

[1] IETF RFC 7915: "IP/ICMP Translation Algorithm".

[2] ISO/IEC 9646-7: "Information technology — Open Systems Interconnection — Conformance testing methodology and framework — Part 7: Implementation Conformance Statements".

[3] IETF RFC 6052: "IPv6 Addressing of IPv4/IPv6 Translators".

[4] IETF RFC 1191: "Path MTU Discovery".

[5] IETF RFC 4884: "Extended ICMP to Support Multi-Part Messages".

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents may be useful in implementing an ETSI deliverable or add to the reader's understanding, but are not required for conformance to the present document.

[i.1] ISO/IEC 9646-1: "Information technology — Open Systems Interconnection — Conformance testing methodology and framework — Part 1: General concepts".

[i.2] ETSI ETS 300 406: "Methods for testing and Specification (MTS); Protocol and profile conformance testing specifications; Standardization methodology".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in IETF RFC 7915 [1] and the following apply:

PICS pro forma: document, in the form of a questionnaire, designed by the protocol specifier or conformance test suite specifier, which, when completed for an OSI implementation or system, becomes the PICS

NOTE: See ISO/IEC 9646-1 [i.1].

Protocol Implementation Conformance Statement (PICS): statement made by the supplier of an Open Systems Interconnection (OSI) implementation or system, stating which capabilities have been implemented for a given OSI protocol

NOTE: See ISO/IEC 9646-1[i.1].

static conformance review: review of the extent to which the static conformance requirements are met by the IUT, accomplished by comparing the PICS with the static conformance requirements expressed in the relevant standard(s)

NOTE: See ISO/IEC 9646-1[i.1].

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in IETF RFC 7915 [1] and the following apply:

3GPP	3rd Generation Partnership Project
AH	Authentication Header
ATS	Abstract Test Suite
DCCP	Datagram Congestion Control Protocol
DF	Don't Fragment
ESP	Encapsulating Security Payload
ICMP	Internet Control Message Protocol
ICMPv4	Internet Control Message Protocol version 4
ICMPv6	Internet Control Message Protocol version 6
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IUT	Implement Under Test
MF	More Fragment
MLD	Multicast Listener Discovery
MTU	Maximum Transport Unit
OSI	Open Systems Interconnection
PDU	Protocol Data Unit
PICS	Protocol Implementation Conformance Statement
PIXIT	Protocol Implementation eXtra Information for Testing
TC	Traffic Class
TCP	Transmission Control Protocol
TOS	Type Of Service
TP	Test Purpose
TSS	Test Suite Structure
TTL	Time To Live
UDP	User Datagram Protocol

4 Conformance

A PICS pro forma which conforms to this PICS pro forma specification shall be technically equivalent to annex A, and shall preserve the numbering and ordering of the items in annex A.

A PICS which conforms to this PICS pro forma specification shall:

a) describe an implementation which claims to conform to IETF RFC 7915 [1];

b) be a conforming ICS pro forma which has been completed in accordance with the instructions for completion given in clause A.1;

c) include the information necessary to uniquely identify both the supplier and the implementation.

Annex A (normative): PICS pro forma

A.1 The right to copy

Notwithstanding the provisions of the copyright clause related to the text of the present document, ETSI grants that users of the present document may freely reproduce the PICS pro forma in this annex so that it can be used for its intended purposes and may further publish the completed PICS pro forma.

A.2 Guidance for completing the PICS pro forma

A.2.1 Purposes and structure

The purpose of this PICS pro forma is to provide a mechanism whereby a supplier of an implementation of the requirements defined in relevant specifications may provide information about the implementation in a standardized manner.

The PICS pro forma is subdivided into clauses for the following categories of information:

- instructions for completing the PICS pro forma;
- identification of the implementation;
- identification of the protocol;
- PICS pro forma tables (for example: Major capabilities, etc.).

A.2.2 Abbreviations and conventions

This annex does not reflect dynamic conformance requirements but static ones. In particular, a condition for support of a PDU parameter does not reflect requirements about the syntax of the PDU (i.e. the presence of a parameter) but the capability of the implementation to support the parameter.

In the sending direction, the support of a parameter means that the implementation is able to send this parameter (but it does not mean that the implementation always sends it).

In the receiving direction, it means that the implementation supports the whole semantic of the parameter that is described in the related protocol specification.

As a consequence, PDU parameter tables in this annex are not the same as the tables describing the syntax of a PDU in the reference specification.

The PICS pro forma contained in this annex is comprised of information in tabular form in accordance with the guidelines presented in ISO/IEC 9646-7.

Item column

The item column contains a number which identifies the item in the table.

Item description column

The item description column describes in free text each respective item (e.g. parameters, timers, etc.). It implicitly means "is <item description> supported by the implementation?".

Reference column

The reference column gives reference to the relevant sections in core specifications.

Status column

The various status used in this annex are in accordance with the rules in table A.1.

Table A.1: Key to status codes

Status code	Status name	Meaning
m	mandatory	The capability shall be supported. It is a static view of the fact that the conformance requirements related to the capability in the reference specification are mandatory requirements. This does not mean that a given behaviour shall always be observed (this would be a dynamic view), but that it shall be observed when the implementation is placed in conditions where the conformance requirements from the reference specification compel it to do so. For instance, if the support for a parameter in a sent PDU is mandatory, it does not mean that it shall always be present, but that it shall be present according to the description of the behaviour in the reference specification (dynamic conformance requirement).
o	optional	The capability may or may not be supported. It is an implementation choice.
n/a	not applicable	It is impossible to use the capability. No answer in the support column is required.
c.<integer>	conditional	The requirement on the capability ("m", "o", "n/a") depends on the support of other optional or conditional items. <integer> is the identifier of the conditional expression.
o.<integer>	qualified optional	For mutually exclusive or selectable options from a set. <integer> is the identifier of the group of options, and the logic of selection of the options.

Mnemonic column

The Mnemonic column contains mnemonic identifiers for each item.

Support column

The support column shall be filled in by the supplier of the implementation. The following common notations, defined in ISO/IEC 9646-7, are used for the support column:

Y or y — supported by the implementation.

N or n — not supported by the implementation.

N/A, n/a or - — no answer required (allowed only if the status is N/A, directly or after evaluation of a conditional status).

References to items

For each possible item answer (answer in the support column) within the PICS pro forma there exists a unique reference, used, for example, in the conditional expressions. It is defined as the table identifier, followed by a solidus character "/", followed by the item number in the table.

EXAMPLE: A.5/4 is the reference to the answer of item 4 in table A.5.

A.2.3 Instructions for completing the PICS pro forma

The supplier of the implementation may complete the PICS pro forma in each of the spaces provided. More detailed instructions are given at the beginning of the different clauses of the PICS pro forma.

A.3 Identification of the Network Equipment

A.3.1 Introduction

Identification of the Network Equipment should be filled in so as to provide as much detail as possible regarding version numbers and configuration options.

The product supplier information and client information should both be filled in if they are different.

A person who can answer queries regarding information supplied in the PICS should be named as the contact person.

A.3.2 Date of the statement

.....

A.3.3 Network Equipment Under Test identification

Name:

.....

.....

Hardware configuration:

.....

.....

.....

Software configuration:

.....

.....

.....

A.3.4 Product supplier

Name:

.....

Address:

.....

.....

.....

Telephone number:

.....

Facsimile number:

.....

E-mail address:

.....

Additional information:

.....

.....

.....

A.3.5 Client

Name:

.....

Address:

.....

.....

.....

Telephone number:

.....

Facsimile number:

.....

E-mail address:

.....

Additional information:

.....

.....

.....

A.3.6 PICS contact person

Name:

.....

Telephone number:

.....

Facsimile number:

.....

E-mail address:

.....

Additional information:

.....

.....

A.4 Identification of the protocol

This PICS pro forma applies to the following specification:

- IETF RFC 7915.

A.5 Global statement of conformance

The implementation described in this PICS meets all the mandatory requirements of the referenced standard?

[ ] Yes

[ ] No

NOTE: Answering "No" to this question indicates non-conformance to the protocol specification. Non-supported mandatory capabilities are to be identified in the PICS, with an explanation of why the implementation is non-conforming. Explanations may be entered in the comments field at the bottom of each table or on attached pages.

In the tabulations which follow, all references are to IETF RFC 7915 unless another numbered reference is explicitly indicated.

A.6 PICS pro forma tables for the IP/ICMP translation algorithm

A.6.1 PICS Items

Table A.2 need only to be completed for implementations.

Table A.2: System Capabilities

Item	Does the IUT support...	Reference	Status	Support
1	Procedures for translating from IPv4 to IPv6?	4	m	
1.1	Procedures for translating IPv4 Headers into IPv6 Headers?	4.1	m	
1.1.1	Configuration function for the network administrator to adjust the threshold of the minimum IPv6 MTU to a value greater than 1 280 bytes if the real value of the minimum IPv6 MTU in the network is known to the administrator?	4.1	m	
1.1.2	Procedures for sending ICMPv4 "Fragmentation Needed" error message to the IPv4 source address If the DF bit is set and the MTU of the next-hop interface is less than the total length value of the IPv4 packet plus 20?	4.1	m	
1.1.3	The IPv6 header contains fields including: Version, Traffic Class, Flow Label, Payload Length, Next Header, Hop Limit, Source Address, Destination Address?	4.1	m	
1.1.3.1	Procedures for setting Version field to 6?	4.1	m	
1.1.3.2	Configurable option to ignore the IPv4 TOS and always set the IPv6 Traffic Class (TC) to zero?	4.1	m	
1.1.3.3	Procedures for setting Flow Label field to all 0?	4.1	m	

Item	Does the IUT support...	Reference	Status	Support
1.1.3.4	Procedures for setting Payload Length field to the value which is total length value from IPv4 header minus the size of the IPv4 header and IPv4 options?	4.1	m	
1.1.3.5	Procedures for copying the protocol field from IPv4 header to Next Header field except that ICMPv4(1) is changed to ICMPv6(58).	4.1	m	
1.1.3.6	Procedures for checking for zero and sending the ICMPv4 "TTL Exceeded" or ICMPv6 "Hop Limit Exceeded" error after decrementing the TTL or Hop Limit.	4.1	m	
1.1.3.7	Procedures for Source Address mapping to an IPv6 address based on the algorithms.	4.1 IETF RFC 6052	m	
1.1.3.7.1	Procedures for silently discarding the packet with an illegal source address (e.g. 0.0.0.0, 127.0.0.1, etc.) except translating ICMPv4 Error Messages into ICMPv6.	4.1	m	
1.1.3.8	Procedures for Destination Address mapping to an IPv6 address based on the algorithms?	4.1 IETF RFC 6052	m	
1.1.4	Procedures for ignoring any options presented in the IPv4 packet?	4.1	m	
1.1.4.1	Procedures for sending an ICMPv4 "Destination Unreachable, Source Route Failed" (Type 3, Code 5) error message when receiving unexpired source route option.	4.1	m	
1.1.5	Procedures for fragmenting when the packet is a fragment or the DF bit is not set and the packet size is greater than the minimum IPv6 MTU in the network set by the translator configuration function.	4.1	m	
1.1.5.1	Procedures for setting Payload Length to the value which is Total length value from the IPv4 header, plus 8 for the Fragment Header, minus the size of the IPv4 header and IPv4 options, if present?	4.1	m	
1.1.5.2	Procedures for setting Next Header field to 44 (Fragment Header)?	4.1	m	
1.1.5.3	Procedures for copying the Next Header field of Fragment Header from IPv4 header to Next Header field except that ICMPv4(1) is changed to ICMPv6(58)?	4.1	m	
1.1.5.4	Procedures for copying Fragment Offset of IPv4 header to Fragment Offset of Fragment Header?	4.1	m	
1.1.5.5	Procedures for copying More Fragment bit of IPv4 header to Fragment Offset of Fragment Header?	4.1	m	
1.1.5.6	Procedures for copying the low-order 16 bits of Identification field of IPv4 header to Identification of Fragment Header and setting high-order 16 bits to zero?	4.1	m	
1.2	Procedures for translating ICMPv4 Headers into ICMPv6 Headers	4.2	m	
1.2.1	Procedures for translating the included IP header in the ICMPv4 error messages?	4.2	m	
1.2.2	Procedures for translating ICMPv4 query messages?	4.2	m	
1.2.2.1	Procedures for adjusting Echo and Echo Reply messages (type 8 and type 0) to the type values to 128 and 129 respectively, and adjusting the ICMP checksum both to take the type change into account and to include the ICMPv6 pseudo-header?	4.2	m	
1.2.2.2	For information Request/Reply messages (Type 15 and Type 16), procedures for silently dropping?	4.2	m	
1.2.2.3	For Timestamp and Timestamp Reply messages (Type 13 and Type 14), procedures for silently dropping?	4.2	m	
1.2.2.4	For Address Mask Request/Reply messages (Type 17 and Type 18), procedures for silently dropping?	4.2	m	
1.2.2.5	For ICMP Router Advertisement messages (Type 9), procedures for silently dropping?	4.2	m	
1.2.2.6	For ICMP Router Solicitation messages (Type 10), procedures for silently dropping?	4.2	m	
1.2.2.7	For Unknown ICMPv4 types messages, procedures for silently dropping?	4.2	m	
1.2.3	Procedures for translating ICMPv4 error messages?	4.2	m	
1.2.3.1	For Destination Unreachable messages (Type 3), procedures for setting the Type to 1 and adjust the ICMP checksum both to take the type/code change into account and to include the ICMPv6 pseudo-header?	4.2	m	
1.2.3.1.1	For code 0 and 1 (Net Unreachable, Host Unreachable), procedures for setting the code to 0 (No route to destination)?	4.2	m	

Item	Does the IUT support...	Reference	Status	Support
1.2.3.1.2	For code 2 (Protocol Unreachable), procedures for translating to an ICMPv6 Parameter Problem (Type 4, Code 1) and making the pointer point to the IPv6 Next Header field?	4.2	m	
1.2.3.1.3	For code 3 (Port Unreachable), procedures for setting the code to 4 (Port unreachable)?	4.2	m	
1.2.3.1.4	For code 4 (Fragmentation Needed and DF was Set), procedures for translating it to an ICMPv6 Packet Too Big message (Type 2) with code set to 0?	4.2	m	
1.2.3.1.4.1	Procedures for adjusting MTU field for the difference between the IPv4 and IPv6 header sizes, and not setting a value smaller than the minimum IPv6 MTU?	4.2	m	
1.2.3.1.4.2	For MTU field being zero, procedures for using the plateau values specified in IETF RFC 1191?	4.2 IETF RFC 1191	m	
1.2.3.1.5	For code 5 (Source Route Failed), procedures for setting the code to 0 (No route to destination)?	4.2	m	
1.2.3.1.6	For code 6, 7, 8, procedures for setting the code to 0 (no route to destination)?	4.2	m	
1.2.3.1.7	For code 9, 10 (Communication with Destination Host Administratively Prohibited), procedures for setting the code to 1 (Communication with destination administratively prohibited)?	4.2	m	
1.2.3.1.8	For code 11, 12, procedures for setting the code to 0 (no route to destination)?	4.2	m	
1.2.3.1.9	For code 13 (Communication Administratively Prohibited), procedures for setting the code to 1 (Communication with destination administratively prohibited)?	4.2	m	
1.2.3.1.10	For code 14 (Host Precedence Violation), procedures for silently dropping?	4.2	m	
1.2.3.1.11	For code 15 (Precedence cutoff in effect), procedures for setting the code to 1 (Communication with destination administratively prohibited)?	4.2	m	
1.2.3.1.12	For other code values, procedures for silently dropping?	4.2	m	
1.2.3.2	For Redirect (Type 5) messages, procedures for silently dropping?	4.2	m	
1.2.3.3	For Alternative Host Address (Type 6) messages, procedures for silently dropping?	4.2	m	
1.2.3.4	For Source Quench (Type 4) messages, procedures for silently dropping?	4.2	m	
1.2.3.5	For Time Exceeded (Type 11) messages, procedures for setting the Type to 3, and adjusting the ICMP checksum both to take the type/code change into account and include the ICMPv6 pseudo-header?	4.2	m	
1.2.3.6	For Parameter Problem (Type 12) messages, procedures for setting the Type to 4, and adjusting the ICMP checksum both to take the type/code change into account and to include the ICMPv6 pseudo-header?	4.2	m	
1.2.3.6.1	For code 0 (Pointer indicates the error), procedures for setting the code to 0(Erroneous header field encountered) and update the pointer as defined in Figure 3 (If the Original IPv4 Pointer Value is not listed or the Translated IPv6 Pointer Value is listed as "n/a", silently drop the packet.)?	4.2 IETF RFC 7915, Figure 3	m	
1.2.3.6.2	For code 1 (Missing a required option), procedures for silently dropping?	4.2	m	
1.2.3.6.3	For code 2 (Bad length), procedures for setting the code to 0 (Erroneous header field encountered) and update the pointer as defined in Figure 3. (If the Original IPv4 Pointer Value is not listed or the Translated IPv6 Pointer Value is listed as "n/a", silently drop the packet.)?	4.2 IETF RFC 7915, Figure 3	m	
1.2.3.6.4	For other code values, procedures for silently dropping?	4.2	m	
1.2.3.7	Procedures for truncating the extension If the ICMPv4 Extension exceeds the maximum size of an ICMPv6 message on the outgoing interface?	4.2	m	
1.3	Procedures for translating ICMPv4 Error Messages into ICMPv6	4.3	m	
1.3.1	Procedures for translating the ICMP error messages containing the packet in error just like a normal IP packet (except the TTL value of the inner IPv4/IPv6 packet)?	4.3	m	

Item	Does the IUT support...	Reference	Status	Support
1.3.2	Procedures for updating the Total Length field in the outer IPv6 header, If the translation of this "packet in error" changes the length of the datagram?	4.3	m	
1.3.3	Procedures for stopping the process of translating the outer IP headers at the first embedded header and dropping the packet if it contains more embedded headers?	4.3	m	
1.4	Generation of ICMPv4 Error Message	4.4	m	
1.4.1	Procedures for sending back an ICMPv4 error message to the original sender of the packet, If the IPv4 packet is discarded, unless the discarded packet is itself an ICMPv4 error message. The ICMPv4 message, if sent, has a Type of 3 (Destination Unreachable) and a Code of 13 (Communication Administratively Prohibited)?	4.4	m	
1.4.2	Function for allowing an administrator to configure whether the ICMPv4 error messages are sent, rate-limited, or not sent.	4.4	m	
1.5	Transport-Layer Header Translation	4.5	m	
1.5.1	Procedures for the recalculation and updating of the transport-layer headers that contain pseudo-headers, If the address translation algorithm is not checksum neutral?	4.5, Section 4.1 of IETF RFC 6052	m	
1.5.2	For UDP packets that do not contain a UDP checksum (i.e. the UDP checksum field is zero), configuration function to allow:1. dropping the packet and generating a system management event that specifies at least the IP addresses and port numbers of the packet. 2. Calculating an IPv6 checksum and forwarding the packet?	4.5	m	
1.5.3	Other transport protocols (e.g. the Datagram Congestion Control Protocol (DCCP)) are OPTIONAL to support.	4.5	o	
1.6	Knowing When to Translate	4.6	m	
1.6.1	If the IP/ICMP translator also provides a normal forwarding function, and the destination IPv4 address is reachable by a more specific route without translation, the procedures for forwarding it without translating it.	4.6	m	
1.6.2	When an IP/ICMP translator receives an IPv4 datagram addressed to an IPv4 destination representing a host in the IPv6 domain, the procedures for translating the packet to IPv6.	4.6	m	
2	Procedures for translating from IPv6 to IPv4?	5	m	
2.1	Procedures for translating IPv6 headers into IPv4 headers?	5.1	m	
2.1.1	If there is no IPv6 Fragment Header, the IPv4 header fields contains Version, Internet Header Length, Type of Service (TOS) Octet, Total Length, Identification, Flags, Fragment Offset, Time to Live, Protocol, Header Checksum, Source Address, Destination address?	5.1	m	
2.1.1.1	Procedures for setting Version field to 4?	5.1	m	
2.1.1.2	Procedures for copying the IPv6 Traffic to the Type of Service (TOS) Octet and providing the ability to ignore the IPv6 traffic class and always set the IPv4 TOS Octet to a specified value?	5.1	m	
2.1.1.3	Procedures for setting the value which is payload length value from the IPv6 header, plus the size of the IPv4 header.	5.1	m	
2.1.1.4	Setting Identification field according to a Fragment Identification generator?	5.1	o	
2.1.1.5	Procedures for setting the More Fragments flag to 0 and the Don't Fragment (DF) flag is set to 0 if the size of the translated IPv4 packet is less than or equal to 1260 bytes; otherwise, it is set to 1?	5.1	m	
2.1.1.6	Procedures for setting Fragment Offset to all zeros?	5.1	m	
2.1.1.7	Procedures for decrementing the Hop Limit value, and checking for 0 and sending the ICMPv4 "TTL Exceeded" or ICMPv6 "Hop Limit Exceeded" error?	5.1	m	
2.1.1.8	Procedures for ignoring IPv6 headers HOPOPT (0), IPv6-Route (43), and IPv6-Opts (60)?	5.1	m	
2.1.1.9	Procedures for computing the header checksum once the IPv4 header has been created?	5.1	m	
2.1.1.10	Procedures for mapping Source address to an IPv4 address based on the algorithms and silently dropping illegal addresses?	5.1	m	
2.1.1.11	Procedures for mapping Destination address to an IPv4 address based on the algorithms?	5.1	m	
2.1.2	Procedures for ignoring any of an IPv6 Hop-by-Hop Options header, Destination Options header, or Routing header with the Segments Left field equal to zero are present in the IPv6 packet?	5.1	m	

Item	Does the IUT support...	Reference	Status	Support
2.1.3	Procedures for returning an ICMPv6 "parameter problem/erroneous header field encountered" (Type 4, Code 0) error message, with the Pointer field indicating the first byte of the Segments Left field", if a Routing header with a non-zero Segments Left field is present?	5.1	m	
2.1.4	Procedures for processing IPv6 fragment?	5.1.1	m	
2.1.4.1	If the Next Header field of the Fragment Header is an extension header (except ESP, but including the Authentication Header (AH)), dropping the packet and log?	5.1.1	m	
2.1.4.2	For other cases, set the Total Length to Payload Length value from IPv6 header, minus the length of the extension headers up to the Fragmentation Header, minus 8 for the Fragment Header, plus the size of the IPv4 header.?	5.1.1	m	
2.1.4.3	Procedures for copying the low-order 16 bits in the Identification field in the Fragment Header to Identification?	5.1.1	m	
2.1.4.4	The IPv4 More Fragments (MF) flag is copied from the M flag in the IPv6 Fragment Header. The IPv4 Don't Fragment (DF) flag is cleared (set to zero), allowing this packet to be further fragmented by IPv4 routers?	5.1.1	m	
2.1.4.5	If the Next Header field of the Fragment Header is not an extension header (except ESP), then Fragment Offset have to be copied from the Fragment Offset field of the IPv6 Fragment Header. If the Next Header field of the Fragment Header is an extension header (except ESP), then the packet should be dropped and logged?	5.1.1	m	
2.1.4.6	For ICMPv6 (58), it is changed to ICMPv4 (1); otherwise, extension headers are skipped, and the Next Header field is copied from the last IPv6 header?	5.1.1	m	
2.1.4.7	If an IPv6 packet that is smaller than or equal to 1280 bytes results (after translation) in an IPv4 packet that is larger than the MTU of the next-hop interface, then the translator has to perform IPv4 fragmentation on that packet.	5.1.1	m	
2.2	Procedures for translating ICMPv6 Headers into ICMPv4 Headers?	5.2	m	
2.2.1	Procedures for translating ICMPv6 informational messages?	5.2	m	
2.2.1.1	For Echo Request and Echo Reply (Type 128 and 129), Adjust the Type values to 8 and 0, respectively, and adjust the ICMP checksum both to take the type change into account and to exclude the ICMPv6 pseudo-header?	5.2	m	
2.2.1.2	For MLD Multicast Listener Query/Report/Done (Type 130, 131, 132), silently drop?	5.2	m	
2.2.1.3	For neighbour Discover messages (Type 133 through 137), silently drop?	5.2	m	
2.2.1.4	For unknown informational messages, silently drop?	5.2	m	
2.2.2	Procedures for ICMPv6 error messages?	5.2	m	
2.2.2.1	For Destination Unreachable (Type 1), set the Type to 3, and adjust the ICMP checksum both to take the type/code change into account and to exclude the ICMPv6 pseudo-header?	5.2	m	
2.2.2.1.1	For Code 0 (No route to destination), set the Code to 1 (Host unreachable)?	5.2	m	
2.2.2.1.2	For Code 1 (Communication with destination administratively prohibited), set the Code to 10 (Communication with destination host administratively prohibited)?	5.2	m	
2.2.2.1.3	For Code 2 (Beyond scope of source address), set the Code to 1 (Host unreachable)?	5.2	m	
2.2.2.1.4	For Code 3 (Address unreachable), set the Code to 1 (Host unreachable)?	5.2	m	
2.2.2.1.5	For Code 4 (Port unreachable), set the Code to 3 (Port unreachable)?	5.2	m	
2.2.2.1.6	For other Code values, silently drop?	5.2	m	
2.2.2.2	For Packet Too Big (Type 2) messages, translate to an ICMPv4 Destination Unreachable (Type 3) with Code 4, and adjust the ICMPv4 checksum both to take the type change into account and to exclude the ICMPv6 pseudo-header?	5.2	m	
2.2.2.2.1	Adjust the MTU field for the difference between the IPv4 and IPv6 header sizes, taking into account whether or not the packet in error includes a Fragment Header, i.e. minimum ((MTU value in the Packet Too Big Message)-20, MTU_of_IPv4_nexthop, (MTU_of_IPv6_nexthop)-20)?	5.2	m	

Item	Does the IUT support...	Reference	Status	Support
2.2.2.3	For Time Exceeded (Type 3) messages, set the Type to 11, and adjust the ICMPv4 checksum both to take the type change into account and to exclude the ICMPv6 pseudo-header?	5.2	m	
2.2.2.4	For Parameter Problem (Type 4) messages, translate the Type and Code and adjust the ICMPv4 checksum both to take the type/code change into account and to exclude the ICMPv6 pseudo-header?	5.2	m	
2.2.2.4.1	For Code 0 (Erroneous header field encountered), set to Type 12, Code 0, and update the pointer as defined in Figure 6. (If the Original IPv6 Pointer Value is not listed or the translated IPv4 Pointer Value is listed as "n/a", silently drop the packet.)?	5.2	m	
2.2.2.4.2	For Code 1 (Unrecognized Next Header type encountered), translate this to an ICMPv4 protocol unreachable (Type 3, Code 2).	5.2	m	
2.2.2.5	For Unknown error messages, silently drop?	5.2	m	
2.2.2.6	If the received ICMPv6 packet contains an ICMPv6 Extension, adjust the ICMPv4 Extension length attribute accordingly?	5.2	m	
2.2.2.7	For extensions not defined in IETF RFC 4884, pass the extensions as opaque bit strings and any IPv6 address literals contained?	5.2 IETF RFC 4884	m	
2.3	Translate ICMPv6 Error Messages into ICMPv4?	5.3	m	
2.3.1	Translate the ICMP error messages containing the packet in error just like a normal IP packet (except that the TTL/Hop Limit value of the inner IPv4/IPv6 packet are not decremented), and update the Total Length field in the outer IPv4 header?	5.3	m	
2.3.2	Translate the inner IP header and stop at the first embedded header and drop the packet if it contains more embedded headers?	5.3	m	
2.4	Generation of ICMPv6 Error Messages?	5.4	m	
2.4.1	If the IPv6 packet is discarded, send back an ICMPv6 error message with Type 1 (Destination Unreachable) and Code 1 (Communication with destination administratively prohibited) to the original sender of the packet, unless the discarded packet is itself an ICMPv6 message?	5.4	m	
2.4.2	Function for allowing an administrator to configure whether the ICMPv6 error messages are sent, rate-limited, or not sent?	5.4	m	
2.5	Transport-Layer Header Translation?	5.5	m	
2.5.1	If the address translation algorithm is not checksum neutral (see Section 4.1 of IETF RFC 6052), perform the recalculation and updating of the transport-layer headers that contain pseudo-headers, and do this for TCP, UDP, and ICMP?	5.5	m	
2.5.2	Support other transport protocols (e.g. DCCP) in order to ease debugging and troubleshooting, and forward all transport protocols as described in the "Protocol" step of Section 5.1.	5	o	
2.6	Knowing when to translate?	5.6	m	
2.6.1	When an IP/ICMP translator receives an IPv6 datagram addressed to an IPv6 address representing a host in the IPv4 domain, translate the IPv6 packet to IPv4?	5.6	m	

History

Version	Date	Status
V1.1.1	February 2026	Publication