ETSI TS 125 470 V15.0.0 (2018-07)

TECHNICAL SPECIFICATION

**Universal Mobile Telecommunications System (UMTS);
UTRAN Iuh Interface PCAP User Adaption (PUA) signalling
(3GPP TS 25.470 version 15.0.0 Release 15)**

Reference

RTS/TSGR-0325470vf00

Keywords

UMTS

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 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the only prevailing document is the print of the Portable Document Format (PDF) version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Copyright Notification

ETSI

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is 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 Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, 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.

Foreword

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under http://webapp.etsi.org/key/queryform.asp.

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.

Contents

Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x   the first digit:

1   presented to TSG for information;

2   presented to TSG for approval;

3   or greater indicates TSG approved document under change control.

y   the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z   the third digit is incremented when editorial only changes have been incorporated in the document.

1   Scope

The present document specifies the *PCAP User Adaption* (PUA) between the Home Node B (HNB) and the Home Node B Gateway (HNB-GW). It fulfils the HNB- HNB-GW communication requirements specified in TS 25.467 [5] and is defined over the Iuh – reference point. It provides transport for PCAP messages.

2   References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.

- For a specific reference, subsequent revisions do not apply.

- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1]   3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2]   3GPP TS 25.453: "UTRAN Iupc interface Positioning Calculation Application Part (PCAP) signalling".

[3]   IETF RFC 4960 (2007-09): "Stream Control Transmission Protocol".

[4]   3GPP TR 25.921 (Version 7.0.0): "Guidelines and principles for protocol description and error handling".

[5]   3GPP TS 25.467: "UTRAN architecture for 3G Home Node B; Stage 2".

[6]   ITU-T Recommendation X.691 (2002-07): "Information technology - ASN.1 encoding rules: Specification of Packed Encoding Rules (PER)".

[7]   ITU-T Recommendation X.680 (2002-07): "Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation".

[8]   ITU-T Recommendation X.681 (2002-07): "Information technology - Abstract Syntax Notation One (ASN.1): Information object specification".

3   Definitions and abbreviations

3.1   Definitions

For the purposes of the present document, the terms and definitions given in TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in TR 21.905 [1].

3.2   Abbreviations

For the purposes of the present document, the abbreviations given in TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in TR 21.905 [1].

CN	Core Network
EP	Elementary Procedure
HNB	Home Node B
HNB-GW	Home Node B Gateway
PDU	Protocol Data Unit
PUA	PCAP User Adaption
SCTP	Stream Control Transmission Protocol

4 General

The protocol described in the present document is the protocol between HNB-GW and HNB.

4.1 Procedure specification principles

The principle for specifying the procedure logic is to specify the functional behaviour of the HNB & HNB-GW exactly and completely.

The following specification principles have been applied for the procedure text in clause 8:

- The procedure text discriminates between:

    1) Functionality which "shall" be executed:

        - The procedure text indicates that the receiving node "shall" perform a certain function Y under a certain condition. If the receiving node supports procedure X but cannot perform functionality Y requested in the REQUEST message of a Class 1 EP, the receiving node shall respond with the message used to report unsuccessful outcome for this procedure, containing an appropriate cause value.

    2) Functionality which "shall, if supported" be executed:

        - The procedure text indicates that the receiving node "shall, if supported," perform a certain function Y under a certain condition. If the receiving node supports procedure X, but does not support functionality Y, the receiving node shall proceed with the execution of the EP, possibly informing the requesting node about the not supported functionality.

- Any required inclusion of an optional IE in a response message is explicitly indicated in the procedure text. If the procedure text does not explicitly indicate that an optional IE shall be included in a response message, the optional IE shall not be included.

4.2 Forwards and backwards compatibility

The forwards and backwards compatibility of the protocol is assured by mechanism where all current and future messages, and IEs or groups of related IEs, include Id and criticality fields that are coded in a standard format that will not be changed in the future. These parts can always be decoded regardless of the standard version.

4.3 Specification notations

For the purposes of the present document, the following notations apply:

Procedure   When referring to an elementary procedure in the specification the Procedure Name is written with the first letters in each word in upper case characters followed by the word "procedure", e.g. Direct Transfer procedure.

Message   When referring to a message in the specification the MESSAGE NAME is written with all letters in upper case characters followed by the word "message", e.g. CONNECT message.

IE   When referring to an information element (IE) in the specification the *Information Element Name* is written with the first letters in each word in upper case characters and all letters in Italic font followed by the abbreviation "IE", e.g. *Context ID* IE.

Value of an IE   When referring to the value of an information element (IE) in the specification the "Value" is written as it is specified in subclause 9.2 enclosed by quotation marks, e.g. "Abstract Syntax Error (Reject)" or "Background ".

5 PUA services

PUA provides the signalling service between the HNB and the HNB-GW that is required to fulfil the PUA functions described in Clause 7.

6 Services expected from the transport layer

Following service is expected from the transport layer:

- reliable and in sequence delivery of Signalling data using SCTP (IETF RFC 4960 [3])

7 Functions of PUA

The PUA has the following functions:

- Transparent transfer of PCAP messages

- Error Handling. This function allows the reporting of general error situations, for which function specific error messages have not been defined.

These functions are implemented by one or several PUA elementary procedures described in the following clauses.

8 PUA procedures

8.1 Elementary Procedures

Table 1 summarizes the EPs.

Table 1; Elementary procedures

Elementary Procedure	Message
Connect	CONNECT
Direct Transfer	DIRECT TRANSFER
Disconnect	DISCONNECT
Connectionless Transfer	CONNECTIONLESS TRANSFER
Error Indication	ERROR INDICATION

8.2 Connect

8.2.1 General

The HNB can initiate this procedure to establish a Signalling Connection and carry a PCAP message.

8.2.2 Successful Operation
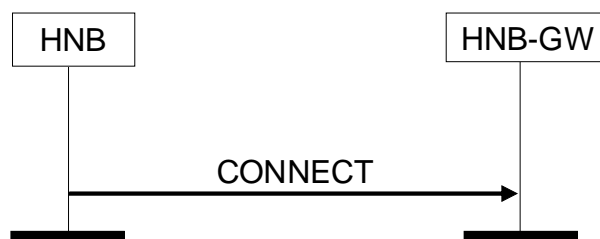
Figure 8.2.2-1: Connect procedure – successful operation

This procedure is used to carry the first PCAP message from the HNB to the HNB-GW.

If the *SAS Indicator* IE is included the HNB-GW may use to select a SAS for connection.

Additional information is provided to enable the HNB-GW to trigger the establishment of a new Signalling Connection between HNB and HNB-GW, which is directly mapped to the Iupc Signalling Connection the PCAP message refers to.

NOTE: The Context ID is used as the Iupc Signalling Connection identifier in the corresponding PCAP messages.

8.3 Direct Transfer

8.3.1 General

This procedure is initiated by either the HNB or HNB-GW to transport a PCAP message between the two nodes.

8.3.2 Successful Operation (HNB-GW Originated)
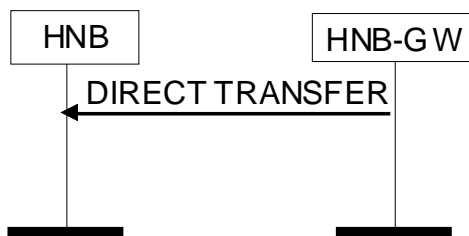
Figure 8.3.2-1: Direct Transfer procedure – successful operation

This procedure is used to carry any downlink connection-oriented PCAP message defined in TS 25.453 [2] from the HNB-GW to the HNB.

8.3.3 Successful Operation (HNB Originated)
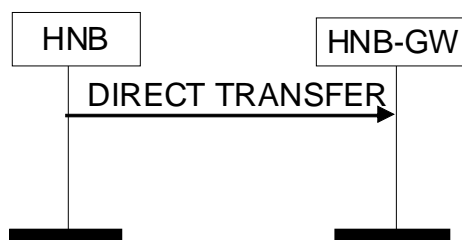
Figure 8.3.3-1: Direct Transfer procedure – successful operation

This procedure is used to carry any uplink connection-oriented PCAP message defined in TS 25.453 [2], except those carried in CONNECT or DISCONNECT messages, from the HNB to the HNB-GW.

8.3.4 Abnormal Conditions

-

8.4 Disconnect

8.4.1 General

This procedure is normally initiated by the HNB to terminate a Signalling Connection between the HNB and HNB-GW, but may be initiated by the HNB-GW to close a refused connection or if an existing connection is released (e.g. for error reasons) by the SAS or HNB-GW.

8.4.2 Successful Operation (HNB Originated)
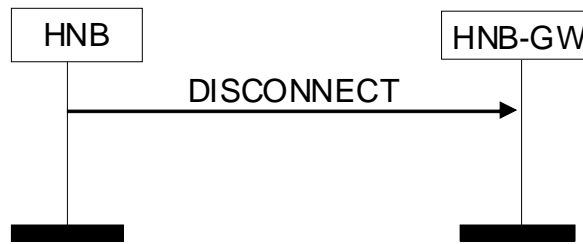
Figure 8.4.2-1: Disconnect procedure

This procedure is used to release a signalling connection between the HNB and HNB-GW and may carry the last PCAP (TS 25.453 [2]) uplink connection-oriented message of a given Signalling Connection to the HNB-GW over the Iuh interface. This procedure may also be used to indicate error conditions at the HNB. This procedure will indicate the cause of the termination in the *Cause* IE.

8.4.3 Successful Operation (HNB-GW Originated)

Figure 8.4.3-1: Disconnect procedure

This procedure is used to close a given Signalling Connection between the HNB and the HNB-GW over the Iuh interface. This procedure will indicate the cause of the termination in the *Cause* IE.

8.5 Connectionless Transfer

8.5.1 General

This procedure is initiated by either the HNB or the HNB-GW to transfer connectionless PCAP messages between the HNB and HNB-GW.

8.5.2 Successful Operation (HNB-GW Originated)

Figure 8.5.2-1: Connectionless Transfer procedure to HNB

This procedure is used to carry any downlink connectionless PCAP message defined in TS 25.453 [2] from the HNB-GW to the HNB.

8.5.3 Successful Operation (HNB Originated)

Figure 8.5.3-1: Connectionless Transfer procedure to HNB-GW

This procedure is used to carry any uplink connectionless PCAP message defined in TS 25.453 [2] from the HNB to the HNB-GW. If the *SAS Indicator* IE is included the HNB-GW may use to select a SAS as the destination of the message.

8.5.4 Abnormal Conditions

-

8.6 Error Indication

8.6.1 General

The Error Indication procedure is initiated by either HNB or HNB-GW to report detected errors in one incoming message.

8.6.2 Successful Operation
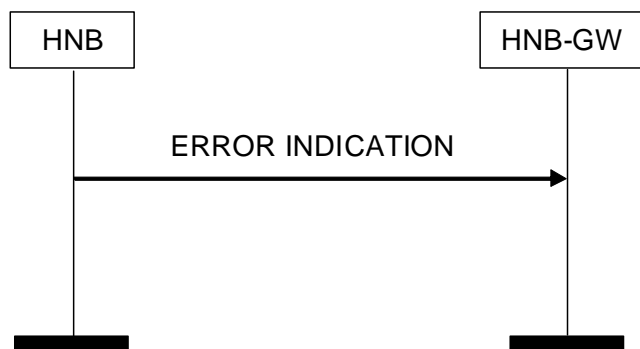
Figure 8.6.2-1 Error Indication HNB Originated, Successful Operation
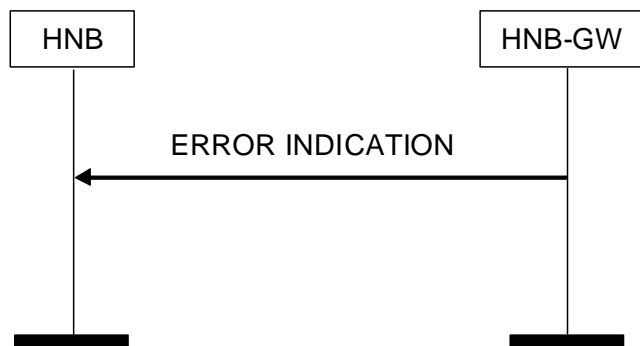
Figure 8.6.2-2 Error Indication HNB-GW Originated, Successful Operation

9 Elements for PUA communication

9.1 Message functional definition and content

9.1.1 General

Section 9.1 presents the contents of PUA messages in tabular format. The corresponding ASN.1 definition is presented in section 9.3. In case there is contradiction between the tabular format in section 9.1 and the ASN.1 definition, the ASN.1 shall take precedence, except for the definition of conditions for the presence of conditional IEs, where the tabular format shall take precedence.

NOTE: The messages have been defined in accordance to the guidelines specified in TR 25.921 [4].

For each message there is, a table listing the signalling elements in their order of appearance in the transmitted message.

9.1.2 Message contents

9.1.2.1 Presence

All information elements in the message descriptions below are marked mandatory, optional or conditional according to table 3.

Table 2: Meaning of abbreviations used in PUA messages

Abbreviation	Meaning
M	IE's marked as Mandatory (M) will always be included in the message.
O	IE's marked as Optional (O) may or may not be included in the message.
C	IE's marked as Conditional (C) will be included in a message only if the condition is satisfied. Otherwise the IE is not included.

9.1.2.2 Criticality

Each Information Element or Group of Information Elements may have a criticality information applied to it. Following cases are possible.

Table 3: Meaning of content within "Criticality" column

Abbreviation	Meaning
–	No criticality information is applied explicitly.
YES	Criticality information is applied. This is usable only for non-repeatable IEs
GLOBAL	The IE and all its repetitions together have one common criticality information. This is usable only for repeatable IEs.
EACH	Each repetition of the IE has its own criticality information. It is not allowed to assign different criticality values to the repetitions. This is usable only for repeatable IEs.

9.1.2.3 Range

The Range column indicates the allowed number of copies of repetitive IEs/IE groups.

9.1.2.4 Assigned Criticality

This column provides the actual criticality information as defined in subclause 10.3.2, if applicable.

9.1.3   CONNECT

This message is sent by the HNB to the HNB-GW to establish a signalling connection and carry a PCAP message.

Direction: HNB → HNB-GW

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		YES	ignore
PCAP Context ID	M		9.2.2		YES	reject
PCAP Message	M		9.2.3		YES	reject
SAS Indicator	O		9.2.4	SAS indicator is used to select a target SAS where multiple SAS are connected and is set by configuration	YES	ignore

9.1.4   DIRECT TRANSFER

This message is sent by either the HNB to the HNB-GW or the HNB-GW to the HNB to transport a connection-oriented PCAP message between the two nodes.

Direction: HNB → HNB-GW and HNB-GW → HNB

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		YES	ignore
PCAP Context ID	M		9.2.2		YES	reject
PCAP Message	M		9.2.3		YES	reject

9.1.5   DISCONNECT

This message is sent either by the HNB to the HNB-GW or the HNB-GW to the HNB to close the signalling connection between the two nodes.

Direction: HNB → HNB-GW and HNB-GW → HNB

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		YES	ignore
PCAP Context ID	M		9.2.2		YES	reject
Cause	M		9.2.6		YES	reject
PCAP Message	O		9.2.3		YES	reject

9.1.6   CONNECTIONLESS TRANSFER

This message is sent by either the HNB to the HNB-GW or the HNB-GW to the HNB to transport a connectionless PCAP message between the two nodes.

Direction: HNB → HNB-GW and HNB-GW → HNB

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		YES	ignore
PCAP Message	M		9.2.3		YES	reject
SAS Indicator	O		9.2.4	SAS indicator is used to select a target SAS where multiple SAS are connected and is set by configuration	YES	ignore

9.1.7   ERROR INDICATION

This message is sent by either the HNB to HNB-GW or the HNB-GW to the HNB and is used to indicate that some errors have been detected.

Direction: HNB → HNB-GW, HNB-GW → HNB

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		YES	ignore
Cause	M		9.2.6		YES	ignore
Criticality Diagnostics	O		9.2.7		YES	ignore
PCAP Context ID	O		9.2.2		YES	ignore
Transaction ID	O		9.2.5		YES	ignore

9.2   Information Element Definitions

9.2.0   General

Section 9.2 presents the PUA IE definitions in tabular format. The corresponding ASN.1 definition is presented in section 9.3. In case there is contradiction between the tabular format in section 9.2 and the ASN.1 definition, the ASN.1 shall take precedence, except for the definition of conditions for the presence of conditional elements, where the tabular format shall take precedence.

When specifying information elements which are to be represented by bitstrings, if not otherwise specifically stated in the semantics description of the concerned IE or elsewhere, the following principle applies with regards to the ordering of bits:

- The first bit (leftmost bit) contains the most significant bit (MSB);

- The last bit (rightmost bit) contains the least significant bit (LSB);

- When importing bitstrings from other specifications, the first bit of the bitstring contains the first bit of the concerned information.

9.2.1   Message Type

Message Type IE uniquely identifies the message being sent. It is mandatory for all messages.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and Reference	Semantics Description
Message Type				
>Procedure Code	M		INTEGER(0..255)	
>Type of Message	M		ENUMERATED (Initiating Message, Successful Outcome, Unsuccessful Outcome, Outcome)	

9.2.2   PCAP Context ID

PCAP Context ID IE uniquely identifies a PUA context in the HNB and HNB-GW.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
Context ID			BIT STRING(SIZE(24))	

9.2.3   PCAP Message

PCAP Message IE contains the transferred PCAP message.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
PCAP Message			OCTET STRING	

9.2.4   SAS Indicator

This element indicates the SAS to be used.

IE/Group Name	Presence	Range	IE type and reference	Semantics description
SAS Indicator	M		INTEGER(0..255)	

9.2.5   Transaction ID

This element indicates the Transaction ID as described in PCAP.

IE/Group Name	Presence	Range	IE Type and Reference	Semantics Description
Transaction ID	M		BIT STRING (SIZE(16))	As defined in PCAP 25.453 [2]

9.2.6   Cause

Cause IE indicates the reason for a particular error event or disconnect for the PUA protocol.

IE/Group Name	Presence	Range	IE Type and Reference	Semantics Description
CHOICE *Cause Group*				
>*Radio Network Layer*				
>>Radio Network Layer Cause	M		ENUMERATED (Normal, Connect failed, HNB-GW release, SAS release, Unspecified, ...,)	
>*Transport Layer*				
>>Transport Layer Cause	M		ENUMERATED (Transport Resource Unavailable, Unspecified, ...)	
>*Protocol*				
>>Protocol Cause	M		ENUMERATED (Transfer Syntax Error, Abstract Syntax Error (Reject), Abstract Syntax Error (Ignore and Notify), Message not Compatible with Receiver State, Semantic Error, Unspecified, Abstract Syntax Error (Falsely Constructed Message), ...)	
>*Misc*				
>>Misc Cause	M		ENUMERATED (Processing Overload, Hardware Failure, O&M Intervention, SAS unavailable, SAS not configured, Unspecified, ...)	

The meaning of the different cause values is described in the following table. Cause values for information 'not valid' indicates that the information is not valid in the context that it was received.

Radio Network Layer cause	Meaning
Normal	No error has occurred
Connect failed	Connect attempt failed
HNB-GW release	Connection released by HNB-GW
SAS release	Connection released by SAS
Unspecified	Sent when none of the above cause values applies but still the cause is Radio Network layer related.

Transport Network Layer cause	Meaning
Transport resource unavailable	**The required transport resources are not available.**
Unspecified	Sent when none of the above cause values applies but still the cause is Transport Network layer related.

Protocol cause	Meaning
Abstract Syntax Error (Reject)	The received message included an abstract syntax error and the concerned criticality indicated "reject".
Abstract Syntax Error (Ignore and Notify)	The received message included an abstract syntax error and the concerned criticality indicated "ignore and notify".
Abstract syntax error (falsely constructed message)	The received message contained IEs in wrong order or with too many occurrences.
Message not Compatible with Receiver State	The received message was not compatible with the receiver state.
Semantic Error	The received message included a semantic error.
Transfer Syntax Error	The received message included a transfer syntax error.
Unspecified	Sent when none of the above cause values applies but still the cause is protocol related.

Miscellaneous cause	Meaning
Processing Overload	Control processing overload.
Hardware Failure	HNB hardware failure.
O&M Intervention	Operation and Maintenance intervention related to HNB.
SAS unavailable	Connection to the SAS is not possible.
SAS not configured	No SAS is configured, or selected SAS is unrecognized.
Unspecified	Sent when none of the above cause values applies.

9.2.7 Criticality Diagnostics

The *Criticality Diagnostics* IE is sent by the HNB or the HNB-GW when parts of a received message have not been comprehended or were missing, or if the message contained logical errors. When applicable, it contains information about which IEs were not comprehended or were missing.

IE/Group Name	Presence	Range	IE type and reference	Semantics description
Criticality Diagnostics				
>Procedure Code	O		INTEGER (0..255)	Procedure Code is to be used if Criticality Diagnostics is part of Error Indication procedure, and not within the response message of the same procedure that caused the error
>Triggering Message	O		ENUMERATED (initiating message, successful outcome, unsuccessful outcome)	The Triggering Message is used only if the Criticality Diagnostics is part of Error Indication procedure.
>Procedure Criticality	O		ENUMERATED(reject, ignore, notify)	This Procedure Criticality is used for reporting the Criticality of the Triggering message (Procedure).
Information Element Criticality Diagnostics		0 to <*maxNrOfErrors*>		
>IE Criticality	M		ENUMERATED(reject, ignore, notify)	The IE Criticality is used for reporting the criticality of the triggering IE. The value 'ignore' shall not be used.
>IE ID	M		INTEGER (0..65535)	The IE Id of the not understood or missing IE

IE/Group Name	Presence	Range	IE type and reference	Semantics description
Criticality Diagnostics				
>Type of Error	M		ENUMERATED(not understood, missing, …)	

Range bound	Explanation
maxNrOfErrors	Maximum no. of IE errors allowed to be reported with a single message. The value for maxNrOfErrors is 256.

9.3 Message and Information Element Abstract Syntax (with ASN.1)

9.3.0 General

PUA ASN.1 definition conforms with ITU-T Rec. X.680 [7] and ITU-T Rec. X.681 [8].

The ASN.1 definition specifies the structure and content of PUA messages. PUA messages can contain any IEs specified in the object set definitions for that message without the order or number of occurrence being restricted by ASN.1. However, for this version of the standard, a sending entity shall construct a PUA message according to the PDU definitions module and with the following additional rules (Note that in the following IE means an IE in the object set with an explicit id. If one IE needed to appear more than once in one object set, then the different occurrences have different IE ids):

- IEs shall be ordered (in an IE container) in the order they appear in object set definitions.

- Object set definitions specify how many times IEs may appear. An IE shall appear exactly once if the presence field in an object has value "mandatory". An IE may appear at most once if the presence field in an object has value "optional" or "conditional". If in a tabular format there is multiplicity specified for an IE (i.e. an IE list) then in the corresponding ASN.1 definition the list definition is separated into two parts. The first part defines an IE container list where the list elements reside. The second part defines list elements. The IE container list appears as an IE of its own. For this version of the standard an IE container list may contain only one kind of list elements.

If a PUA message that is not constructed as defined above is received, this shall be considered as Abstract Syntax Error, and the message shall be handled as defined for Abstract Syntax error in subclause 10.3.6.

9.3.1 Usage of private message mechanism for non-standard use

The private message mechanism for non-standard use may be used:

- for special operator- (and/or vendor) specific features considered not to be part of the basic functionality, i.e. the functionality required for a complete and high-quality specification in order to guarantee multivendor interoperability;

- by vendors for research purposes, e.g. to implement and evaluate new algorithms/features before such features are proposed for standardisation.

The private message mechanism shall not be used for basic functionality. Such functionality shall be standardised.

9.3.2 Elementary Procedure Definitions

```
-- **************************************************************
--
-- Elementary Procedure definitions
--
-- **************************************************************

PUA-PDU-Descriptions  {
itu-t (0) identified-organization (4) etsi (0) mobileDomain (0)
umts-Access (20) modules (3) pua(7) version1 (1) pUA-PDU-Descriptions (0)}
```

```
DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

-- **************************************************************
--
-- IE parameter types from other modules.
--
-- **************************************************************

IMPORTS
    Criticality,
    ProcedureCode
FROM PUA-CommonDataTypes
    Connect,
    DirectTransfer,
    Disconnect,
    ConnectionlessTransfer,
    ErrorIndication,
    PrivateMessage


FROM PUA-PDU-Contents
    id-Connect,
    id-DirectTransfer,
    id-Disconnect,
    id-ConnectionlessTransfer,
    id-ErrorIndication,
    id-privateMessage
FROM PUA-Constants;

-- **************************************************************
--
-- Interface Elementary Procedure Class
--
-- **************************************************************

PUA-ELEMENTARY-PROCEDURE ::= CLASS {
    &InitiatingMessage          ,
    &SuccessfulOutcome          OPTIONAL,
    &UnsuccessfulOutcome        OPTIONAL,
    &procedureCode              ProcedureCode   UNIQUE,
    &criticality                Criticality     DEFAULT ignore
}

WITH SYNTAX {
    INITIATING MESSAGE          &InitiatingMessage
    [SUCCESSFUL OUTCOME         &SuccessfulOutcome]
    [UNSUCCESSFUL OUTCOME       &UnsuccessfulOutcome]
    PROCEDURE CODE              &procedureCode
    [CRITICALITY                &criticality]
}

-- **************************************************************
```

ETSI

```
--
-- Interface PDU definitions
--
-- *************************************************************

PUA-PDU ::= CHOICE {
    initiatingMessage      InitiatingMessage,
    successfulOutcome      SuccessfulOutcome,
    unsuccessfulOutcome    UnsuccessfulOutcome,
    ...
}


InitiatingMessage ::= SEQUENCE {
    procedureCode   PUA-ELEMENTARY-PROCEDURE.&procedureCode      ({PUA-ELEMENTARY-PROCEDURES}),
    criticality     PUA-ELEMENTARY-PROCEDURE.&criticality        ({PUA-ELEMENTARY-PROCEDURES}{@procedureCode}),
    value           PUA-ELEMENTARY-PROCEDURE.&InitiatingMessage ({PUA-ELEMENTARY-PROCEDURES}{@procedureCode})
}

SuccessfulOutcome ::= SEQUENCE {
    procedureCode   PUA-ELEMENTARY-PROCEDURE.&procedureCode      ({PUA-ELEMENTARY-PROCEDURES}),
    criticality     PUA-ELEMENTARY-PROCEDURE.&criticality        ({PUA-ELEMENTARY-PROCEDURES}{@procedureCode}),
    value           PUA-ELEMENTARY-PROCEDURE.&SuccessfulOutcome ({PUA-ELEMENTARY-PROCEDURES}{@procedureCode})
}

UnsuccessfulOutcome ::= SEQUENCE {
    procedureCode   PUA-ELEMENTARY-PROCEDURE.&procedureCode        ({PUA-ELEMENTARY-PROCEDURES}),
    criticality     PUA-ELEMENTARY-PROCEDURE.&criticality          ({PUA-ELEMENTARY-PROCEDURES}{@procedureCode}),
    value           PUA-ELEMENTARY-PROCEDURE.&UnsuccessfulOutcome  ({PUA-ELEMENTARY-PROCEDURES}{@procedureCode})
}

-- *************************************************************
--
-- Interface Elementary Procedure List
--
-- *************************************************************

PUA-ELEMENTARY-PROCEDURES PUA-ELEMENTARY-PROCEDURE ::= {
    PUA-ELEMENTARY-PROCEDURES-CLASS-1    |
    PUA-ELEMENTARY-PROCEDURES-CLASS-2    ,
    ...
}

PUA-ELEMENTARY-PROCEDURES-CLASS-1 PUA-ELEMENTARY-PROCEDURE ::= {
    ...
}


PUA-ELEMENTARY-PROCEDURES-CLASS-2 PUA-ELEMENTARY-PROCEDURE ::= {
    connectionRequest |
    directTransfer |
    disconnectRequest |
    connectionlessTransfer |
    errorIndication |
```

```
    privateMessage,
    ...
}

-- *************************************************************
--
-- Interface Elementary Procedures
--
-- *************************************************************


connectionRequest PUA-ELEMENTARY-PROCEDURE ::= {
    INITIATING MESSAGE      Connect
    PROCEDURE CODE          id-Connect
    CRITICALITY             ignore
}

directTransfer PUA-ELEMENTARY-PROCEDURE ::= {
    INITIATING MESSAGE      DirectTransfer
    PROCEDURE CODE          id-DirectTransfer
    CRITICALITY             ignore
}

disconnectRequest PUA-ELEMENTARY-PROCEDURE ::= {
    INITIATING MESSAGE      Disconnect
    PROCEDURE CODE          id-Disconnect
    CRITICALITY             ignore
}
connectionlessTransfer PUA-ELEMENTARY-PROCEDURE ::= {
    INITIATING MESSAGE      ConnectionlessTransfer
    PROCEDURE CODE          id-ConnectionlessTransfer
    CRITICALITY             ignore
}

errorIndication PUA-ELEMENTARY-PROCEDURE ::= {
    INITIATING MESSAGE      ErrorIndication
    PROCEDURE CODE          id-ErrorIndication
    CRITICALITY             ignore
}

privateMessage PUA-ELEMENTARY-PROCEDURE ::= {
    INITIATING MESSAGE      PrivateMessage
    PROCEDURE CODE          id-privateMessage
    CRITICALITY             ignore
}


END
```

9.3.3   PDU definitions

```
-- *************************************************************
--
```

```
-- PDU definitions for PUA.
--
-- *************************************************************

PUA-PDU-Contents {
itu-t (0) identified-organization (4) etsi (0) mobileDomain (0)
umts-Access (20) modules (3) pua(7) version1 (1) pUA-PDU-Contents (1) }

DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

-- *************************************************************
--
-- IE parameter types from other modules.
--
-- *************************************************************

IMPORTS
    Cause,
    CriticalityDiagnostics,
    PCAP-Context-ID,
    PCAP-Message,
    SAS-Indicator,
    TransactionID

FROM PUA-IEs


    ProtocolExtensionContainer{},
    ProtocolIE-ContainerList{},
    ProtocolIE-Container{},
    ProtocolIE-Single-Container{},
    PrivateIE-Container{},
    PUA-PRIVATE-IES,
    PUA-PROTOCOL-EXTENSION,
    PUA-PROTOCOL-IES
FROM PUA-Containers

    id-Cause,
    id-CriticalityDiagnostics,
    id-PCAP-Context-ID,
    id-PCAP-Message,
    id-SAS-Indicator,
    id-TransactionID

FROM PUA-Constants;


-- *************************************************************
--
-- Connect
--
-- *************************************************************
```

```
Connect ::= SEQUENCE {
    protocolIEs         ProtocolIE-Container        { {ConnectIEs} },
    protocolExtensions  ProtocolExtensionContainer { {ConnectExtensions} }  OPTIONAL,
    ...
}

ConnectIEs PUA-PROTOCOL-IES ::= {
    { ID id-PCAP-Context-ID             CRITICALITY reject  TYPE PCAP-Context-ID        PRESENCE mandatory } |
    { ID id-PCAP-Message                CRITICALITY reject  TYPE PCAP-Message           PRESENCE mandatory } |
    { ID id-SAS-Indicator               CRITICALITY ignore  TYPE SAS-Indicator          PRESENCE optional },
    ...
}

ConnectExtensions PUA-PROTOCOL-EXTENSION ::= {
    ...
}


-- *************************************************************
--
-- Direct Transfer
--
-- *************************************************************

DirectTransfer ::= SEQUENCE {
    protocolIEs         ProtocolIE-Container        { {DirectTransferIEs} },
    protocolExtensions  ProtocolExtensionContainer { {DirectTransferExtensions} }  OPTIONAL,
    ...
}

DirectTransferIEs PUA-PROTOCOL-IES ::= {
    { ID id-PCAP-Context-ID             CRITICALITY reject  TYPE PCAP-Context-ID        PRESENCE mandatory } |
    { ID id-PCAP-Message                CRITICALITY reject  TYPE PCAP-Message           PRESENCE mandatory  },
    ...
}

DirectTransferExtensions PUA-PROTOCOL-EXTENSION ::= {
    ...
}


-- *************************************************************
--
-- Disconnect
--
-- *************************************************************

Disconnect ::= SEQUENCE {
    protocolIEs         ProtocolIE-Container        { {DisconnectIEs} },
    protocolExtensions  ProtocolExtensionContainer { {DisconnectExtensions} }  OPTIONAL,
    ...
}
```

```
DisconnectIEs PUA-PROTOCOL-IES ::= {
    { ID id-PCAP-Context-ID              CRITICALITY reject  TYPE PCAP-Context-ID      PRESENCE mandatory } |
    { ID id-Cause                        CRITICALITY reject  TYPE Cause                PRESENCE mandatory } |
    { ID id-PCAP-Message                 CRITICALITY reject  TYPE PCAP-Message         PRESENCE optional  },
    ...
}

DisconnectExtensions PUA-PROTOCOL-EXTENSION ::= {
    ...
}


-- *************************************************************
--
-- Connectionless Transfer
--
-- *************************************************************

ConnectionlessTransfer ::= SEQUENCE {
    protocolIEs        ProtocolIE-Container       { {ConnectionlessTransferIEs} },
    protocolExtensions  ProtocolExtensionContainer { {ConnectionlessTransferExtensions} }   OPTIONAL,
    ...
}

ConnectionlessTransferIEs PUA-PROTOCOL-IES ::= {
    { ID id-PCAP-Message                 CRITICALITY reject  TYPE PCAP-Message         PRESENCE mandatory } |
    { ID id-SAS-Indicator                CRITICALITY ignore  TYPE SAS-Indicator        PRESENCE optional },
    ...
}

ConnectionlessTransferExtensions PUA-PROTOCOL-EXTENSION ::= {
    ...
}


-- *************************************************************
--
-- ERROR INDICATION
--
-- *************************************************************

ErrorIndication ::= SEQUENCE {
    protocolIEs        ProtocolIE-Container       { {ErrorIndicationIEs} },
    protocolExtensions  ProtocolExtensionContainer { {ErrorIndicationExtensions} }  OPTIONAL,
    ...
}

ErrorIndicationIEs PUA-PROTOCOL-IES ::= {
    { ID id-Cause                        CRITICALITY ignore  TYPE Cause                    PRESENCE mandatory } |
    { ID id-CriticalityDiagnostics       CRITICALITY ignore  TYPE CriticalityDiagnostics   PRESENCE optional } |
    { ID id-PCAP-Context-ID              CRITICALITY ignore  TYPE PCAP-Context-ID          PRESENCE optional } |
    { ID id-TransactionID                CRITICALITY ignore  TYPE TransactionID            PRESENCE optional },
     ...
}
```

```
ErrorIndicationExtensions PUA-PROTOCOL-EXTENSION ::= {
    ...
}

-- ***************************************************************
--
-- PRIVATE MESSAGE
--
-- ***************************************************************

PrivateMessage ::= SEQUENCE {
    privateIEs      PrivateIE-Container  {{PrivateMessage-IEs}},
    ...
}

PrivateMessage-IEs PUA-PRIVATE-IES ::= {
    ...
}


END
```

9.3.4   Information Element definitions

```
-- ***************************************************************
--
-- Information Element Definitions
--
-- ***************************************************************

PUA-IEs {
itu-t (0) identified-organization (4) etsi (0) mobileDomain (0)
umts-Access (20) modules (3) pua(7) version1 (1) pUA-IEs (2) }

DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

IMPORTS
    maxNrOfErrors
FROM PUA-Constants

    Criticality,
    ProcedureCode,
    ProtocolIE-ID,
    TriggeringMessage
FROM PUA-CommonDataTypes

    ProtocolExtensionContainer{},
    PUA-PROTOCOL-EXTENSION
```

```
FROM PUA-Containers;


PCAP-Context-ID ::=  BIT STRING (SIZE(24))


PCAP-Message    ::=          OCTET STRING

SAS-Indicator   ::=          INTEGER(0..255)


TransactionID        ::= BIT STRING (SIZE(16))



-- ************************************************************
--
-- Cause IE
--
-- ************************************************************

Cause ::= CHOICE {
    radioNetwork            CauseRadioNetwork,
    transport               CauseTransport,
    protocol                CauseProtocol,
    misc                    CauseMisc,
    ...
}
CauseRadioNetwork ::= ENUMERATED {
    normal,
    connect-failed,
    hnb-gw-release,
    sas-release,
    unspecified,
    ...
}
CauseTransport ::= ENUMERATED {
    transport-resource-unavailable,
    unspecified,
    ...
}

CauseProtocol ::= ENUMERATED {
    transfer-syntax-error,
    abstract-syntax-error-reject,
    abstract-syntax-error-ignore-and-notify,
    message-not-compatible-with-receiver-state,
    semantic-error,
    unspecified,
    abstract-syntax-error-falsely-constructed-message,
    ...
}

CauseMisc ::= ENUMERATED {
```

```
    processing-overload,
    hardware-failure,
    o-and-m-intervention,
    sas-unavailable,
    sas-not-configured,
    unspecified,
    ...
}


-- *************************************************************
--
-- CriticalityDiagnostics
--
-- *************************************************************

CriticalityDiagnostics ::= SEQUENCE {
    procedureCode               ProcedureCode                                           OPTIONAL,
    triggeringMessage           TriggeringMessage                                       OPTIONAL,
    procedureCriticality        Criticality                                             OPTIONAL,
    iEsCriticalityDiagnostics   CriticalityDiagnostics-IE-List                          OPTIONAL,
    iE-Extensions               ProtocolExtensionContainer { {CriticalityDiagnostics-ExtIEs} }   OPTIONAL,
    ...
}

CriticalityDiagnostics-IE-List ::= SEQUENCE (SIZE (1..maxNrOfErrors)) OF
    SEQUENCE {
        iECriticality           Criticality,
        iE-ID                   ProtocolIE-ID,
        typeOfError             TypeOfError,
        iE-Extensions           ProtocolExtensionContainer { {CriticalityDiagnostics-IE-List-ExtIEs} } OPTIONAL,
        ...
    }

CriticalityDiagnostics-IE-List-ExtIEs PUA-PROTOCOL-EXTENSION ::= {
    ...
}

CriticalityDiagnostics-ExtIEs PUA-PROTOCOL-EXTENSION ::= {
    ...
}


TypeOfError ::= ENUMERATED {
    not-understood,
    missing,
    ...
}

END
```

9.3.5   Common definitions

```
-- *************************************************************
--
-- Common definitions
--
-- *************************************************************

PUA-CommonDataTypes {
itu-t (0) identified-organization (4) etsi (0) mobileDomain (0)
umts-Access (20) modules (3) pua(7) version1 (1) pUA-CommonDataTypes (3)  }

DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

-- *************************************************************
--
-- Extension constants
--
-- *************************************************************

maxPrivateIEs                              INTEGER ::= 65535
maxProtocolExtensions                      INTEGER ::= 65535
maxProtocolIEs                             INTEGER ::= 65535


-- *************************************************************
--
-- Common Data Types
--
-- *************************************************************
Criticality     ::= ENUMERATED { reject, ignore, notify }

Presence        ::= ENUMERATED { optional, conditional, mandatory }


ProcedureCode       ::= INTEGER (0..255)

PrivateIE-ID    ::= CHOICE {
    local               INTEGER (0..65535),
    global              OBJECT IDENTIFIER
}

ProtocolIE-ID       ::= INTEGER (0..maxProtocolIEs)

TriggeringMessage   ::= ENUMERATED { initiating-message, successful-outcome, unsuccessful-outcome }

END
```

9.3.6   Constant definitions

```
-- *************************************************************
--
```

```
-- Constant definitions
--
-- *************************************************************

PUA-Constants {
itu-t (0) identified-organization (4) etsi (0) mobileDomain (0)
umts-Access (20) modules (3) pua(7) version1 (1) pUA-Constants (4) }

DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

IMPORTS
    ProcedureCode,
    ProtocolIE-ID
FROM PUA-CommonDataTypes;


-- *************************************************************
--
-- Elementary Procedures
--
-- *************************************************************

id-Connect                          ProcedureCode ::= 1
id-DirectTransfer                   ProcedureCode ::= 2
id-Disconnect                       ProcedureCode ::= 3
id-ConnectionlessTransfer           ProcedureCode ::= 4
id-ErrorIndication                  ProcedureCode ::= 5
id-privateMessage                   ProcedureCode ::= 6

-- *************************************************************
--
-- Lists
--
-- *************************************************************

maxNrOfErrors                       INTEGER ::= 256

-- *************************************************************
--
-- IEs
--
-- *************************************************************

id-Cause                                ProtocolIE-ID ::= 1
id-CriticalityDiagnostics               ProtocolIE-ID ::= 2
id-PCAP-Context-ID                      ProtocolIE-ID ::= 3
id-PCAP-Message                         ProtocolIE-ID ::= 4
id-SAS-Indicator                        ProtocolIE-ID ::= 5
id-TransactionID                        ProtocolIE-ID ::= 6

END
```

9.3.7   Container definitions

```
-- *************************************************************
--
-- Container definitions
--
-- *************************************************************

PUA-Containers {
itu-t (0) identified-organization (4) etsi (0) mobileDomain (0)
umts-Access (20) modules (3) pua(7) version1 (1) pUA-Containers (5) }

DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

-- *************************************************************
--
-- IE parameter types from other modules.
--
-- *************************************************************

IMPORTS
    Criticality,
    Presence,
    PrivateIE-ID,
    ProtocolIE-ID,
    maxPrivateIEs,
    maxProtocolExtensions,
    maxProtocolIEs
FROM PUA-CommonDataTypes;

-- *************************************************************
--
-- Class Definition for Private IEs
--
-- *************************************************************

PUA-PRIVATE-IES ::= CLASS {
    &id               PrivateIE-ID,
    &criticality      Criticality,
    &Value,
    &presence         Presence
}
WITH SYNTAX {
    ID                &id
    CRITICALITY       &criticality
    TYPE              &Value
    PRESENCE          &presence
}
```

ETSI

```
-- *************************************************************
--
-- Class Definition for Protocol IEs
--
-- *************************************************************

PUA-PROTOCOL-IES ::= CLASS {
    &id                 ProtocolIE-ID       UNIQUE,
    &criticality        Criticality,
    &Value,
    &presence           Presence
}
WITH SYNTAX {
    ID                  &id
    CRITICALITY         &criticality
    TYPE                &Value
    PRESENCE            &presence
}

-- *************************************************************
--
-- Class Definition for Protocol Extensions
--
-- *************************************************************

PUA-PROTOCOL-EXTENSION ::= CLASS {
    &id                 ProtocolIE-ID UNIQUE,
    &criticality        Criticality,
    &Extension,
    &presence           Presence
}
WITH SYNTAX {
    ID                  &id
    CRITICALITY         &criticality
    EXTENSION           &Extension
    PRESENCE            &presence
}

-- *************************************************************
--
-- Container for Protocol IEs
--
-- *************************************************************

ProtocolIE-Container {PUA-PROTOCOL-IES : IEsSetParam} ::=
    SEQUENCE (SIZE (0..maxProtocolIEs)) OF
        ProtocolIE-Field {{IEsSetParam}}

ProtocolIE-Single-Container {PUA-PROTOCOL-IES : IEsSetParam} ::=
    ProtocolIE-Field {{IEsSetParam}}

ProtocolIE-Field {PUA-PROTOCOL-IES : IEsSetParam} ::= SEQUENCE {
```

```
    id                  PUA-PROTOCOL-IES.&id                    ({IEsSetParam}),
    criticality         PUA-PROTOCOL-IES.&criticality           ({IEsSetParam}{@id}),
    value               PUA-PROTOCOL-IES.&Value                 ({IEsSetParam}{@id})
}

-- *************************************************************
--
-- Container Lists for Protocol IE Containers
--
-- *************************************************************

ProtocolIE-ContainerList {INTEGER : lowerBound, INTEGER : upperBound, PUA-PROTOCOL-IES : IEsSetParam} ::=
    SEQUENCE (SIZE (lowerBound..upperBound)) OF
        ProtocolIE-Container {{IEsSetParam}}

-- *************************************************************
--
-- Container for Protocol Extensions
--
-- *************************************************************

ProtocolExtensionContainer {PUA-PROTOCOL-EXTENSION : ExtensionSetParam} ::=
    SEQUENCE (SIZE (1..maxProtocolExtensions)) OF
        ProtocolExtensionField {{ExtensionSetParam}}

ProtocolExtensionField {PUA-PROTOCOL-EXTENSION : ExtensionSetParam} ::= SEQUENCE {
    id                  PUA-PROTOCOL-EXTENSION.&id                   ({ExtensionSetParam}),
    criticality         PUA-PROTOCOL-EXTENSION.&criticality          ({ExtensionSetParam}{@id}),
    extensionValue      PUA-PROTOCOL-EXTENSION.&Extension            ({ExtensionSetParam}{@id})
}


-- *************************************************************
--
-- Container for Private IEs
--
-- *************************************************************

PrivateIE-Container {PUA-PRIVATE-IES : IEsSetParam } ::=
    SEQUENCE (SIZE (1.. maxPrivateIEs)) OF
        PrivateIE-Field {{IEsSetParam}}

PrivateIE-Field {PUA-PRIVATE-IES : IEsSetParam} ::= SEQUENCE {
    id                  PUA-PRIVATE-IES.&id                   ({IEsSetParam}),
    criticality         PUA-PRIVATE-IES.&criticality          ({IEsSetParam}{@id}),
    value               PUA-PRIVATE-IES.&Value                ({IEsSetParam}{@id})
}


END
```

ETSI

9.4   Message transfer syntax

PUA shall use the ASN.1 Basic Packed Encoding Rules (BASIC-PER) Aligned Variant as transfer syntax as specified in ref. ITU-T Rec. X.691 [6].

10 Handling of unknown, unforeseen, and erroneous protocol data

10.1 General

Protocol Error cases can be divided into three classes:

- Transfer Syntax Error;

- Abstract Syntax Error;

- Logical Error.

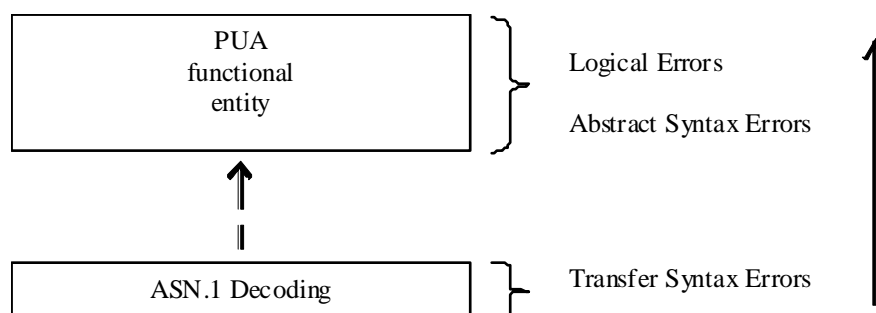Protocol errors can occur in the following functions within a receiving node:

Figure 10.1-1: Protocol errors in PUA

The information stated in subclauses 10.2, 10.3 and 10.4, to be included in the message used when reporting an error, is what at minimum shall be included. Other optional information elements within the message may also be included, if available. This is also valid for the case when the reporting is done with a response message. The latter is an exception to what is stated in subclause 4.1.

10.2 Transfer Syntax Error

A Transfer Syntax Error occurs when the receiver is not able to decode the received physical message Transfer syntax errors are always detected in the process of ASN.1 decoding. If a Transfer Syntax Error occurs, the receiver should initiate Error Indication procedure with appropriate cause value for the Transfer Syntax protocol error.

10.3 Abstract Syntax Error

10.3.1 General

An Abstract Syntax Error occurs when the receiving functional PUA entity:

1. receives IEs or IE groups that cannot be understood (unknown IE id);

2. receives IEs for which the logical range is violated (e.g.: ASN.1 definition: 0 to 15, the logical range is 0 to 10 (values 11 to 15 are undefined), and 12 will be received; this case will be handled as an abstract syntax error using criticality information sent by the originator of the message);

3. does not receive IEs or IE groups but according to the specified presence of the concerning object, the IEs or IE groups should have been present in the received message;

4. receives IEs or IE groups that are defined to be part of that message in wrong order or with too many occurrences of the same IE or IE group;

5. receives IEs or IE groups but according to the conditional presence of the concerning object and the specified condition, the IEs or IE groups should not have been present in the received message.

Cases 1 and 2 (not comprehended IE/IE group) are handled based on received Criticality information. Case 3 (missing IE/IE group) is handled based on Criticality information and Presence information for the missing IE/IE group specified in the version of the specification used by the receiver. Case 4 (IEs or IE groups in wrong order or with too many occurrences) and Case 5 (erroneously present conditional IEs or IE groups) result in rejecting the procedure.

If an Abstract Syntax Error occurs, the receiver shall read the remaining message and shall then for each detected Abstract Syntax Error act according to the Criticality Information and Presence Information for the IE/IE group due to which Abstract Syntax Error occurred in accordance with subclauses 10.3.4 and 10.3.5. The handling of cases 4 and 5 is specified in subclause 10.3.6.

10.3.2   Criticality Information

In the PUA messages there is criticality information set for individual IEs and/or IE groups. This criticality information instructs the receiver how to act when receiving an IE or an IE group that is not comprehended i.e. the entire item (IE or IE group) which is not (fully or partially) comprehended shall be treated in accordance with its own criticality information as specified in subclause 10.3.4.

In addition, the criticality information is used in case of the missing IE/IE group abstract syntax error (see subclause 10.3.5).

The receiving node shall take different actions depending on the value of the Criticality Information. The three possible values of the Criticality Information for an IE/IE group are:

- Reject IE;

- Ignore IE and Notify Sender;

- Ignore IE.

The following rules restrict when a receiving entity may consider an IE, an IE group or an EP not comprehended (not implemented), and when action based on criticality information is applicable:

1. IE or IE group: When one new or modified IE or IE group is implemented for one EP from a standard version, then other new or modified IEs or IE groups specified for that EP in that standard version shall be considered comprehended by the receiving entity (some may still remain unsupported).

2. EP: The comprehension of different EPs within a standard version or between different standard versions is not mandated. Any EP that is not supported may be considered not comprehended, even if another EP from that standard version is comprehended, and action based on criticality shall be applied.

10.3.3   Presence Information

For many IEs/IE groups which are optional according to the ASN.1 transfer syntax, PUA specifies separately if the presence of these IEs/IE groups is optional or mandatory with respect to RNS application by means of the presence field of the concerning object of class PUA-PROTOCOL-IES, PUA-PROTOCOL-IES-PAIR, PUA-PROTOCOL-EXTENSION or PUA-PRIVATE-IES.

The presence field of the indicated classes supports three values:

1. Optional;

2. Conditional;

3. Mandatory.

If an IE/IE group is not included in a received message and the presence of the IE/IE group is mandatory or the presence is conditional and the condition is true according to the version of the specification used by the receiver, an abstract syntax error occurs due to a missing IE/IE group.

10.3.4   Not comprehended IE/IE group

10.3.4.1   Procedure Code

The receiving node shall treat the different types of received criticality information of the *Procedure Code* according to the following:

Reject IE:

- If a message is received with a *Procedure Code* marked with "*Reject IE*" which the receiving node does not comprehend, the receiving node shall reject the procedure using the Error Indication procedure.

Ignore IE and Notify Sender:

- If a message is received with a *Procedure Code* marked with "*Ignore IE and Notify Sender*" which the receiving node does not comprehend, the receiving node shall ignore the procedure and initiate the Error Indication procedure.

Ignore IE:

- If a message is received with a *Procedure Code* marked with "*Ignore IE*" which the receiving node does not comprehend, the receiving node shall ignore the procedure.

When using the Error Indication procedure to reject a procedure or to report an ignored procedure it shall include the *Procedure Code* IE, the *Triggering Message* IE, and the *Procedure Criticality* IE in the *Criticality Diagnostics* IE.

10.3.4.1A   Type of Message

When the receiving node cannot decode the *Type of Message* IE, the Error Indication procedure shall be initiated with an appropriate cause value.

10.3.4.2   IEs other than the Procedure Code and Type of Message

The receiving node shall treat the different types of received criticality information of an IE/IE group other than the *Procedure Code* IE and *Type of Message* IE according to the following:

Reject IE:

- If a message *initiating* a procedure is received containing one or more IEs/IE groups marked with "*Reject IE*" which the receiving node does not comprehend; none of the functional requests of the message shall be executed. The receiving node shall reject the procedure and report the rejection of one or more IEs/IE groups using the message normally used to report unsuccessful outcome of the procedure. In case the information received in the initiating message was insufficient to determine a value for all IEs that are required to be present in the message used to report the unsuccessful outcome of the procedure, the receiving node shall instead terminate the procedure and initiate the Error Indication procedure.

- If a message *initiating* a procedure that does not have a message to report unsuccessful outcome is received containing one or more IEs/IE groups marked with "*Reject IE*" which the receiving node does not comprehend, the receiving node shall terminate the procedure and initiate the Error Indication procedure.

- If a *response* message is received containing one or more IEs marked with "*Reject IE*" which the receiving node does no comprehend, the receiving node shall consider the procedure as unsuccessfully terminated and initiate local error handling.

Ignore IE and Notify Sender:

- If a message *initiating* a procedure is received containing one or more Ies/IE groups marked with "*Ignore IE and Notify Sender*" which the receiving node does not comprehend, the receiving node shall ignore the content of the not comprehended IEs/IE groups, continue with the procedure as if the not comprehended IEs/IE groups were not received (except for the reporting) using the understood IEs/IE groups, and report in the response message of the procedure that one or more IEs/IE groups have been ignored. In case the information received in the initiating message was insufficient to determine a value for all IEs that are required to be present in the response message, the receiving node shall instead terminate the procedure and initiate the Error Indication procedure.

- if a message *initiating* a procedure that does not have a message to report the outcome of the procedure is received containing one or more IEs/IE groups marked with "*Ignore IE and Notify Sender*" which the receiving node does not comprehend, the receiving node shall ignore the content of the not comprehended IEs/IE groups, continue with the procedure as if the not comprehended IEs/IE groups were not received (except for the reporting) using the understood IEs/IE groups, and initiate the Error Indication procedure to report that one or more IEs/IE groups have been ignored.

- If a *response* message is received containing one or more IEs/IE groups marked with "*Ignore IE and Notify Sender*" which the receiving node does not comprehend, the receiving node shall ignore the content of the not comprehended IE/IE groups, continue with the procedure as if the not comprehended IEs/IE groups were not received (except for the reporting) using the understood IEs/IE groups and initiate the Error Indication procedure.

Ignore IE:

- If a message *initiating* a procedure is received containing one or more IEs/IE groups marked with "*Ignore IE*" which the receiving node does not comprehend, the receiving node shall ignore the content of the not comprehended IEs/IE groups and continue with the procedure as if the not comprehended IEs/IE groups were not received using only the understood IEs/IE groups.

- If a *response* message is received containing one or more IEs/IE groups marked with "*Ignore IE*" which the receiving node does not comprehend, the receiving node shall ignore the content of the not comprehended IEs/IE groups and continue with the procedure as if the not comprehended IEs/IE groups were not received using the understood IEs/IE groups.

When reporting not comprehended IEs/IE groups marked with "*Reject IE*" or "*Ignore IE and Notify Sender*" using a response message defined for the procedure, the *Information Element Criticality Diagnostics* IE shall be included in the *Criticality Diagnostics* IE for each reported IE/IE group.

When reporting not comprehended IEs/IE groups marked with "*Reject IE*" or "*Ignore IE and Notify Sender*" using the Error Indication procedure, the *Procedure Code* IE, the *Triggering Message* IE, *Procedure Criticality* IE, and the *Information Element Criticality Diagnostics* IE shall be included in the *Criticality Diagnostics* IE for each reported IE/IE group.

10.3.5   Missing IE or IE group

The receiving node shall treat the missing IE/IE group according to the criticality information for the missing IE/IE group in the received message specified in the version of the present document used by the receiver:

Reject IE:

- if a received message *initiating* a procedure is missing one or more IEs/IE groups with specified criticality "*Reject IE*"; none of the functional requests of the message shall be executed. The receiving node shall reject the procedure and report the missing IEs/IE groups using the message normally used to report unsuccessful outcome of the procedure. In case the information received in the initiating message was insufficient to determine a value for all IEs that are required to be present in the message used to report the unsuccessful outcome of the procedure, the receiving node shall instead terminate the procedure and initiate the Error Indication procedure.

- if a received message *initiating* a procedure that does not have a message to report unsuccessful outcome is missing one or more IEs/IE groups with specified criticality "*Reject IE*", the receiving node shall terminate the procedure and initiate the Error Indication procedure.

- if a received *response* message is missing one or more IEs/IE groups with specified criticality "*Reject IE*, the receiving node shall consider the procedure as unsuccessfully terminated and initiate local error handling.

Ignore IE and Notify Sender:

- if a received message *initiating* a procedure is missing one or more IEs/IE groups with specified criticality "*Ignore IE and Notify Sender*", the receiving node shall ignore that those IEs are missing and continue with the procedure based on the other IEs/IE groups present in the message and report in the response message of the procedure that one or more IEs/IE groups were missing. In case the information received in the initiating message was insufficient to determine a value for all IEs that are required to be present in the response message, the receiving node shall instead terminate the procedure and initiate the Error Indication procedure.

- if a received message *initiating* a procedure that does not have a message to report the outcome of the procedure is missing one or more IEs/IE groups with specified criticality "*Ignore IE and Notify Sender*", the receiving node shall ignore that those IEs are missing and continue with the procedure based on the other IEs/IE groups present in the message and initiate the Error Indication procedure to report that one or more IEs/IE groups were missing.

- if a received *response* message is missing one or more IEs/IE groups with specified criticality "*Ignore IE and Notify Sender*", the receiving node shall ignore that those IEs are missing and continue with the procedure based on the other IEs/IE groups present in the message and initiate the Error Indication procedure to report that one or more IEs/IE groups were missing.

Ignore IE:

- if a received message *initiating* a procedure is missing one or more IEs/IE groups with specified criticality "*Ignore IE*", the receiving node shall ignore that those IEs are missing and continue with the procedure based on the other IEs/IE groups present in the message.

- if a received *response* message is missing one or more IEs/IE groups with specified criticality "*Ignore IE*", the receiving node shall ignore that those IEs/IE groups are missing and continue with the procedure based on the other IEs/IE groups present in the message.

When reporting missing IEs/IE groups with specified criticality "*Reject IE*" or "*Ignore IE and Notify Sender*" using a response message defined for the procedure, the *Information Element Criticality Diagnostics* IE shall be included in the *Criticality Diagnostics* IE for each reported IE/IE group.

When reporting missing IEs/IE groups with specified criticality "*Reject IE*" or "*Ignore IE and Notify Sender*" using the Error Indication procedure, the *Procedure Code* IE, the *Triggering Message* IE, *Procedure Criticality* IE, and the *Information Element Criticality Diagnostics* IE shall be included in the *Criticality Diagnostics* IE for each reported IE/IE group.

10.3.6 IEs or IE groups received in wrong order or with too many occurrences or erroneously present

If a message with IEs or IE groups in wrong order or with too many occurrences is received or if IEs or IE groups with a conditional presence are present when the condition is not met (i.e. erroneously present), the receiving node shall behave according to the following:

- If a message *initiating* a procedure is received containing IEs or IE groups in wrong order or with too many occurrences or erroneously present, none of the functional requests of the message shall be executed. The receiving node shall reject the procedure and report the cause value "Abstract Syntax Error (Falsely Constructed Message)" using the message normally used to report unsuccessful outcome of the procedure. In case the information received in the initiating message was insufficient to determine a value for all IEs that are required to be present in the message used to report the unsuccessful outcome of the procedure, the receiving node shall instead terminate the procedure and initiate the Error Indication procedure.

- If a message *initiating* a procedure that does not have a message to report unsuccessful outcome is received containing IEs or IE groups in wrong order or with too many occurrences or erroneously present, the receiving node shall terminate the procedure and initiate the Error Indication procedure, and use cause value "Abstract Syntax Error (Falsely Constructed Message)".

- If a *response* message is received containing IEs or IE groups in wrong order or with too many occurrences or erroneously present, the receiving node shall consider the procedure as unsuccessfully terminated and initiate local error handling.

When determining the correct order only the IEs specified in the specification version used by the receiver shall be considered.

10.4　Logical Error

Logical error situations occur when a message is comprehended correctly, but the information contained within the message is not valid (i.e. semantic error), or describes a procedure which is not compatible with the state of the receiver. In these conditions, the following behaviour shall be performed (unless otherwise specified) as defined by the class of the elementary procedure, irrespective of the criticality information of the IE's/IE groups containing the erroneous values.

Class 1:

Where the logical error occurs in a request message of a class 1 procedure, and the procedure has a message to report this unsuccessful outcome, this message shall be sent with an appropriate cause value. Typical cause values are:

- Semantic Error;

- Message not compatible with receiver state.

Where the logical error is contained in a request message of a class 1 procedure, and the procedure does not have a message to report this unsuccessful outcome, the procedure shall be terminated and the Error Indication procedure shall be initiated with an appropriate cause value. The *Procedure Code* IE and the *Triggering Message* IE within the *Criticality Diagnostics* IE shall then be included in order to identify the message containing the logical error.

Where the logical error exists in a response message of a class 1 procedure, the procedure shall be considered as unsuccessfully terminated and local error handling shall be initiated.

Class 2:

Where the logical error occurs in a message of a class 2 procedure, the procedure shall be terminated and the Error Indication procedure shall be initiated with an appropriate cause value. The *Procedure Code* IE and the *Triggering Message* IE within the *Criticality Diagnostics* IE shall then be included in order to identify the message containing the logical error.

10.5　Exceptions

The error handling for all the cases described hereafter shall take precedence over any other error handling described in the other subclauses of clause 10.

- If any type of error (Transfer Syntax Error, Abstract Syntax Error or Logical Error) is detected in the ERROR INDICATION message, it shall not trigger the Error Indication procedure in the receiving Node but local error handling.

- In case a response message or Error Indication message needs to be returned, but the information necessary to determine the receiver of that message is missing, the procedure shall be considered as unsuccessfully terminated and local error handling shall be initiated.

- If an error that terminates a procedure occurs, the returned cause value shall reflect the error that caused the termination of the procedure even if one or more abstract syntax errors with criticality "ignore and notify" have earlier occurred within the same procedure.

Annex A (informative): Change history

Change history								
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**		**Old**	**New**
14-11-2013	RAN3#82	R3-132041			Skeleton		0.0.0	0.0.1
14-11-2013	RAN3#82	R3-132042			Prelim Material added		0.0.1	0.1.0
14-11-2013	RAN3#82	R3-132393			Corrected prelim material, added EPs.		0.1.0	0.2.0
14-11-2013	RAN3#82	R3-132395			Added IEs and ASN.1		0.2.0	0.3.0
22-11-2013	RAN3#82	R3-132394			Updated		0.3.0	0.4.0
26-11-2013	RAN#62	RP-131649			Presented for approval		0.4.0	1.0.0
03-12-2013	RAN#62				Corrected templates		1.0.0	1.0.1
12-2013					Upgraded to Rel-12, and put under change control		1.0.1	12.0.0
06-2014	RAN#64	RP-140907	0001	-	Add the Object Identifier for PUA		12.0.0	12.1.0
12-2014	RAN#66	RP-142094	0002	-	Rapporteur Review		12.1.0	12.2.0
12-2015	RAN#70				Update to Rel-13 version (MCC)		12.2.0	13.0.0
03-2016	RAN#71	RP-160449	0004		Rapporteur Review of TS 25.470		13.0.0	13.1.0

Change history							
Date	**Meeting**	**TDoc**	**CR**	**Rev**	**Cat**	**Subject/Comment**	**New version**
2017-03	SA#75					Promotion to Release 14 without technical change	14.0.0
2018-07	SA#80	-	-	-	-	Promotion to Release 15 without technical change	15.0.0

History

Document history		
V15.0.0	July 2018	Publication