

**3GPP2 C.S0084-004-0**

**Version 3.0**

**Date: August 2008**



**3RD GENERATION  
PARTNERSHIP  
PROJECT 2  
"3GPP2"**

---

## ***Application Layer for Ultra Mobile Broadband (UMB) Air Interface Specification***

**COPYRIGHT**

*3GPP2 and its Organizational Partners claim copyright in this document and individual Organizational Partners may copyright and issue documents or standards publications in individual Organizational Partner's name based on this document. Requests for reproduction of this document should be directed to the 3GPP2 Secretariat at <mailto:secretariat@3gpp2.org>. Requests to reproduce individual Organizational Partner's documents should be directed to that Organizational Partner. See <http://www.3gpp2.org/> for more information.*

No text.

**CONTENTS**

1	FOREWORD .....	xi
2	NOTES .....	xiii
3	REFERENCES .....	xiii
4	1 Introduction .....	1-1
5	2 Basic Signaling Protocol .....	2-1
6	2.1 General Overview .....	2-1
7	2.2 Primitives and Public Data .....	2-1
8	2.2.1 Commands .....	2-1
9	2.2.2 Return Indications .....	2-1
10	2.2.3 Local Common Data .....	2-1
11	2.2.4 Public Data .....	2-1
12	2.3 Protocol Data Unit .....	2-1
13	2.4 General Signaling Requirements .....	2-2
14	2.4.1 General Requirements .....	2-2
15	2.4.2 Message Information .....	2-2
16	2.5 Procedures and Messages for the InConfiguration Instance of the Protocol .....	2-3
17	2.5.1 Protocol Initialization for the InConfiguration Protocol Instance .....	2-3
18	2.5.2 Procedures .....	2-4
19	2.5.3 Message Formats .....	2-4
20	2.6 Procedures and Messages for the InUse Instance of the Protocol .....	2-4
21	2.6.1 Protocol Initialization for the InUse Protocol Instance .....	2-4
22	2.6.2 Hard Commit Procedures .....	2-4
23	2.6.3 Soft Commit Procedures .....	2-4
24	2.6.4 Procedures .....	2-4
25	2.6.5 Type Definitions .....	2-4
26	2.6.6 Messages .....	2-5
27	2.6.7 BSP Header .....	2-5
28	2.7 Interface to Other Protocols .....	2-5
29	2.7.1 Commands .....	2-5
30	2.7.2 Indications .....	2-5
31	2.8 Configuration Attributes .....	2-5

**CONTENTS**

1	2.9 Non-Attribute Data .....	2-5
2	2.10 Session State Information .....	2-5
3	3 Basic Inter-Route Tunneling Protocol.....	3-1
4	3.1 Overview .....	3-1
5	3.2 Primitives, Local Common Data, and Public Data.....	3-2
6	3.2.1 Commands.....	3-2
7	3.2.2 Return Indications .....	3-2
8	3.2.3 Local Common Data .....	3-2
9	3.2.4 Public Data .....	3-2
10	3.3 Protocol Data Unit.....	3-2
11	3.4 Procedures and Messages for the InConfiguration Instance of the Protocol.....	3-3
12	3.4.1 Procedures .....	3-3
13	3.4.1.1 Protocol Initialization.....	3-3
14	3.4.2 Message Formats .....	3-3
15	3.5 Procedures and Messages for the InUse Instance of the Protocol.....	3-3
16	3.5.1 Procedures .....	3-3
17	3.5.1.1 Protocol Initialization.....	3-3
18	3.5.1.2 Hard Commit Procedures .....	3-3
19	3.5.1.3 Soft Commit Procedures .....	3-4
20	3.5.1.4 Inter-Route Tunneling Protocol Header.....	3-4
21	3.5.1.5 Command Processing .....	3-6
22	3.5.1.5.1 Activate .....	3-6
23	3.5.1.5.2 Deactivate.....	3-6
24	3.5.1.6 Inter-Route Tunneling Protocol Stream Mapping Procedures .....	3-6
25	3.5.1.7 Transmission Procedures .....	3-6
26	3.5.1.7.1 Access Terminal Procedure .....	3-6
27	3.5.1.7.2 Access Network Procedures .....	3-7
28	3.5.1.8 Reception Procedures .....	3-7
29	3.5.1.8.1 Access Terminal Procedures .....	3-7
30	3.5.1.8.2 Access Network Procedures .....	3-8
31	3.5.2 Type Definitions .....	3-8
32	3.5.3 Message Formats .....	3-8

**CONTENTS**

1 3.6 Interface to Other Protocols .....3-9

2 3.6.1 Commands Sent .....3-9

3 3.6.2 Indications .....3-9

4 3.7 Configuration Attributes .....3-9

5 3.7.1 Simple Attributes .....3-9

6 3.7.2 Complex Attributes .....3-9

7 3.8 Non-Attribute Data .....3-9

8 3.9 Session State Information .....3-9

9 4 Basic ROHC Support Protocol .....4-1

10 4.1 Overview .....4-1

11 4.2 Primitives, Local Common Data, and Public Data .....4-1

12 4.2.1 Commands .....4-1

13 4.2.2 Return Indications .....4-1

14 4.2.3 Local Common Data .....4-1

15 4.2.4 Public Data .....4-1

16 4.3 Protocol Data Unit .....4-1

17 4.4 Protocol Initialization .....4-1

18 4.5 Procedures and Messages for the InConfiguration Instance of the Protocol .....4-1

19 4.5.1 Protocol Initialization for the InConfiguration Protocol Instance .....4-1

20 4.5.2 Procedures .....4-1

21 4.5.3 Message Formats .....4-1

22 4.6 Procedures and Messages for the InUse Instance of the Protocol .....4-2

23 4.6.1 Procedures .....4-2

24 4.6.1.1 Protocol Initialization for the InUse Protocol Instance .....4-2

25 4.6.1.2 Hard Commit Procedures .....4-2

26 4.6.1.3 Soft Commit Procedures .....4-2

27 4.6.1.4 ROHC Support Protocol Header .....4-2

28 4.6.1.5 Command Processing .....4-2

29 4.6.1.6 Common Procedures .....4-2

30 4.6.1.6.1 Use of Multiple ROHC Versions .....4-2

31 4.6.2 Type Definitions .....4-2

32 4.6.3 Message Formats .....4-3

**CONTENTS**

1	4.7 Interface to Other Protocols.....	4-3
2	4.7.1 Commands Sent.....	4-3
3	4.7.2 Indications .....	4-3
4	4.8 Configuration Attributes .....	4-3
5	4.8.1 Simple Attributes .....	4-3
6	4.8.2 Complex Attributes .....	4-3
7	4.8.2.1 ROHCPParametersFwd Attribute .....	4-3
8	4.8.2.2 ROHCPParametersRev Attribute .....	4-4
9	4.9 Non-Attribute Data .....	4-6
10	4.10 Session State Information .....	4-6
11	5 Basic EAP Support Protocol .....	5-1
12	5.1 Overview .....	5-1
13	5.2 Primitives, Local Common Data, and Public Data.....	5-1
14	5.2.1 Commands.....	5-1
15	5.2.2 Return Indications .....	5-1
16	5.2.3 Local Common Data .....	5-1
17	5.2.4 Public Data .....	5-1
18	5.3 Protocol Data Unit.....	5-1
19	5.4 Protocol Initialization .....	5-1
20	5.5 Procedures and Messages for the InConfiguration Instance of the Protocol.....	5-1
21	5.5.1 Protocol Initialization for the InConfiguration Protocol Instance.....	5-1
22	5.5.2 Procedures .....	5-1
23	5.5.3 Message Formats .....	5-2
24	5.6 Procedures and Messages for the InUse Instance of the Protocol.....	5-2
25	5.6.1 Procedures .....	5-2
26	5.6.1.1 Protocol Initialization for the InUse Protocol Instance .....	5-2
27	5.6.1.2 Hard Commit Procedures .....	5-2
28	5.6.1.3 Soft Commit Procedures .....	5-2
29	5.6.1.4 EAP Support Protocol Header .....	5-2
30	5.6.1.5 Command Processing .....	5-2
31	5.6.1.6 Common Procedures .....	5-2
32	5.6.1.7 Access Terminal Procedures .....	5-2

**CONTENTS**

1	5.6.1.8 Access Network Procedures.....	5-3
2	5.6.2 Type Definitions.....	5-3
3	5.6.3 Message Formats.....	5-3
4	5.6.3.1 InitiateEAPFastReauthentication .....	5-3
5	5.6.3.2 InitiateEAP.....	5-3
6	5.7 Interface to Other Protocols .....	5-4
7	5.7.1 Commands Sent .....	5-4
8	5.7.2 Indications .....	5-4
9	5.8 Configuration Attributes.....	5-4
10	5.8.1 Simple Attributes.....	5-4
11	5.8.2 Complex Attributes.....	5-4
12	5.9 Non-Attribute Data.....	5-4
13	5.10 Session State Information.....	5-4
14		

**CONTENTS**

- 1 No text.

**FIGURES**

1 Figure 2-1. BSP Packet Structure.....2-2

2 Figure 2-2. Sample Message Information.....2-3

3 Figure 3-1. Data Path for Normal and Tunneled Packets .....3-1

4 Figure 3-2. Example Inter-Route Tunneling Protocol Headers .....3-2

5 Figure 3-3. Inter-Route Tunneling Protocol Packet .....3-3

6

**FIGURES**

- 1 No text.

**TABLES**

1 Table 3-1. HeaderType Values .....3-4  
2 Table 3-2. SpecialRouteID Values.....3-6  
3 Table 3-3. Configurable Values.....3-9  
4 Table 5-1. Configurable Values.....5-4  
5

**TABLES**

- 1 No text.

**FOREWORD**

1 **(This foreword is not part of this Standard)**

2 This Standard was prepared by Technical Specification Group C of the Third Generation  
3 Partnership Project 2 (3GPP2). This Standard is the Application Layer part of the Ultra  
4 Mobile Broadband™ (UMB™)<sup>1</sup> air interface. Other parts of this Standard are:

- 5 • Overview for Ultra Mobile Broadband (UMB) Air Interface Specification
- 6 • Physical Layer for Ultra Mobile Broadband (UMB) Air Interface Specification
- 7 • MAC Layer for Ultra Mobile Broadband (UMB) Air Interface Specification
- 8 • Radio Link Layer for Ultra Mobile Broadband (UMB) Air Interface Specification
- 9 • Security Functions for Ultra Mobile Broadband (UMB) Air Interface Specification
- 10 • Connection Control Plane for Ultra Mobile Broadband (UMB) Air Interface Specification
- 11 • Session Control Plane for Ultra Mobile Broadband (UMB) Air Interface Specification
- 12 • Route Control Plane for Ultra Mobile Broadband (UMB) Air Interface Specification
- 13 • Broadcast-Multicast Upper Layers for Ultra Mobile Broadband (UMB) Air Interface  
14 Specification

15 Other Standards may be required to implement this system and are listed in the References  
16 section of each part.

17 This standard provides a specification for land mobile wireless systems based upon cellular  
18 principles. This Standard is one part of the IMT-2000 CDMA Multi-Carrier, IMT-2000  
19 CDMA MC, also known as cdma2000®<sup>2</sup>.

---

<sup>1</sup> Ultra Mobile Broadband™ and (UMB™) are trade and service marks owned by the CDMA Development Group (CDG).

<sup>2</sup> cdma2000® is the trademark for the technical nomenclature for certain specifications and standards of the Organizational Partners (OPs) of 3GPP2. Geographically (and as of the date of publication), cdma2000® is a registered trademark of the Telecommunications Industry Association (TIA-USA) in the United States.

**FOREWORD**

- 1 No text.

**REFERENCES**

1 The following documents contain provisions, which, through reference in this text,  
2 constitute provisions of this document. References are either specific (identified by date of  
3 publication, edition number, version number, etc.) or non-specific. For a specific reference,  
4 subsequent revisions do not apply. For a non-specific reference, the latest version applies.  
5 In the case of a reference to a 3GPP2 document, a non-specific reference implicitly refers to  
6 the latest version of that document in the same Release as the present document.

- 7
- 8 [1] C.S0084-000-0, Overview for Ultra Mobile Broadband (UMB) Air Interface  
9 Specification.
- 10 [2] C.S0084-001-0, Physical Layer for Ultra Mobile Broadband (UMB) Air Interface  
11 Specification.
- 12 [3] C.S0084-002-0, MAC Layer for Ultra Mobile Broadband (UMB) Air Interface  
13 Specification.
- 14 [4] C.S0084-003-0, Radio Link Layer for Ultra Mobile Broadband (UMB) Air Interface  
15 Specification.
- 16 [5] Reserved.
- 17 [6] C.S0084-005-0, Security Functions for Ultra Mobile Broadband (UMB) Air Interface  
18 Specification.
- 19 [7] C.S0084-006-0, Connection Control Plane for Ultra Mobile Broadband (UMB) Air  
20 Interface Specification.
- 21 [8] C.S0084-007-0, Session Control Plane for Ultra Mobile Broadband (UMB) Air  
22 Interface Specification.
- 23 [9] C.S0084-008-0, Route Control Plane for Ultra Mobile Broadband (UMB) Air  
24 Interface Specification.
- 25 [10] C.S0084-009-0, Broadcast-Multicast Upper Layer for Ultra Mobile Broadband  
26 (UMB) Air Interface Specification.
- 27 [11] C.R1001, Administration of Parameter Value Assignments for cdma2000 Spread  
28 Spectrum Standards. (Informative)
- 29 [12] IETF RFC 3095, Robust Header Compression (ROHC): Framework and four  
30 profiles: RTP, UDP, ESP, and uncompressed.
- 31 [13] RObust Header Compression (ROHC) Profile Identifiers  
32 [<http://www.iana.org/assignments/rohcn-pro-ids>]
- 33 [14] RFC 3748, Extensible Authentication Protocol (EAP).

34

## **REFERENCES**

- 1 No text.

1 **1 INTRODUCTION**

2 Application Layer consists of following protocols:

- 3 • Basic Signaling Protocol
- 4 • Basic Inter-Route Tunneling Protocol
- 5 • Basic ROHC Support Protocol
- 6 • Basic EAP Support Protocol

- 1 No text.

## 1 **2 BASIC SIGNALING PROTOCOL**

### 2 **2.1 General Overview**

3 Protocols in each layer use Basic Signaling Protocol (BSP) to exchange messages.

4 BSP provides a one or two octet header that defines the Type of the protocol with which the  
5 message is associated. The BSP uses the header to route the message to the appropriate  
6 protocol instance.

7 BSP is a message-routing protocol, and routes messages to protocols specified by the Type  
8 field provided in the BSP header. Protocol type is assigned to InUse as well as  
9 InConfiguration instance of each protocol (see [1]).

10 The actual protocol indicated by the Type is negotiated during session set-up. For example,  
11 Type 0x07 is associated with the Reverse Control Channel MAC Protocol. The specific  
12 Reverse Control Channel MAC Protocol used (and, therefore, the Reverse Control Channel  
13 MAC protocol generating and processing the messages delivered by BSP) is negotiated  
14 when the session is setup.

15 The remainder of the message following the Type field (BSP header) is processed by the  
16 protocol specified by the Type.

### 17 **2.2 Primitives and Public Data**

#### 18 2.2.1 Commands

19 This protocol does not define any commands.

#### 20 2.2.2 Return Indications

21 This protocol returns the following indications:

- 22 • *ReliableMessageDeliveryFailed*
- 23 • *ReliableSignalingStreamReset*

#### 24 2.2.3 Local Common Data

25 This protocol does not define any Local Common Data.

#### 26 2.2.4 Public Data

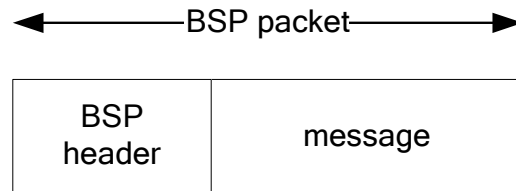
27 This protocol shall make the following data public:

- 28 • All data defined as Static Attribute, Static Non-Attribute Data, and Local Common Data

### 29 **2.3 Protocol Data Unit**

30 The protocol data unit for this protocol is a BSP packet. Each BSP packet consists of one  
31 message sent by a protocol using BSP.

32 The protocol constructs a BSP packet by adding the BSP header (see 2.6.7) in front of the  
33 payload. The structure of the BSP packet is shown in Figure 2-1.



1

2

**Figure 2-1. BSP Packet Structure**

### 3 **2.4 General Signaling Requirements**

#### 4 2.4.1 General Requirements

5 The following requirements are common to all protocols that carry messages using BSP and  
6 that provide for message extensibility. The access terminal and the access network shall  
7 abide by the following rules when generating and processing any signaling message carried  
8 by BSP.

- 9 • Messages shall always be an integer number of octets in length; and, if necessary,  
10 include a Reserved field in the message to make them so. The receiver shall ignore the  
11 value of the Reserved fields.
- 12 • The first field of the message shall be transmitted first. Within each field, the most  
13 significant bit of the field shall be transmitted first.
- 14 • Message identifiers for messages that cannot be sent using broadcast addressing shall  
15 be unambiguous for each protocol (Type, Subtype) pair.
- 16 • Message identifiers for messages that may be sent using broadcast addressing shall be  
17 unambiguous for each protocol Type.
- 18 • For future revisions, the transmitter shall add new fields only at the end of a message  
19 (excluding any Reserved field). The transmitter shall not add fields if their addition  
20 makes the parsing of previous fields ambiguous for receivers.
- 21 • The receiver shall discard all unrecognized messages.
- 22 • The receiver shall discard all unrecognized fields.
- 23 • The receiver shall discard a message if any of the fields in the message is set to a value  
24 outside of the defined field range, unless the receiver is specifically directed to ignore  
25 this field. A field value is outside of the allowed range if a range was specified with the  
26 field and the value is not in this range, or the field is set to a value that is defined as  
27 invalid. Values that are designated as reserved are considered to be outside of the  
28 allowed range.

#### 29 2.4.2 Message Information

30 Each message definition contains information regarding channels on which the message  
31 can be transmitted, whether the message requires RLP reliable or best-effort delivery, the  
32 addressing modes applicable to the message, and whether AuthTag is required in the

1 message. This information is provided in the form of a table, an example of which is given  
 2 in Figure 2-2.

<b>Channels</b>	FTC    RTC	<b>RLP</b>	Best Effort
<b>Addressing</b>	Unicast	<b>AuthTag</b>	Required

4 **Figure 2-2. Sample Message Information**

5 The following values are defined:

- 6 • **Channels:** This information field indicates the Physical Layer Channels on which this  
 7 message can be transmitted. The sender of the message shall send the message only on  
 8 the Physical Layer channel(s) indicated by this information field. Values are:
  - 9 – FTC for Forward Unicast Traffic Channel,
  - 10 – F-BCMCSCH for Forward Broadcast Multicast Services Channel
  - 11 – RTC for Reverse Traffic Channel,
- 12 • **RLP:** Radio Link Protocol requirements. The sender of the message shall send the  
 13 message only using the RLP in the mode(s) indicated by this information field. Values  
 14 are:
  - 15 – Best Effort: the message is sent once and is subject to erasure, and
  - 16 – Reliable: erasures are detected and the message is retransmitted one or more times,  
 17 if necessary.
- 18 • **Addressing:** Addressing modes for the message. The sender of the message shall send  
 19 the message only with an address type(s) indicated by this information field. Values are:
  - 20 – Broadcast if a broadcast address can be used with this message, and
  - 21 – Unicast if a unicast address can be used with this message.
- 22 • **AuthTag:** AuthTag requirements. The sender of the message shall send the message  
 23 with AuthTag requirements indicated by this information field. Values are:
  - 24 – Required: if this message shall not be sent without an AuthTag,
  - 25 – Not Required: if an AuthTag is not required with this message, and
  - 26 – Required when key is available: if an AuthTag shall be sent with this message when  
 27 Message Integrity Key for the channel carrying this message is available (e.g.,  
 28 FACMIKey[KeyIndex] is not set to NULL). See [6].

## 29 **2.5 Procedures and Messages for the InConfiguration Instance of the Protocol**

### 30 2.5.1 Protocol Initialization for the InConfiguration Protocol Instance

31 Upon creation, the InConfiguration instance of this protocol in the access terminal and the  
 32 access network shall perform the procedures specified in [1].

## 1 2.5.2 Procedures

2 This protocol uses the services of the Session Control Protocol to perform negotiation of  
3 attribute values.

## 4 2.5.3 Message Formats

5 This protocol does not define any messages.

## 6 **2.6 Procedures and Messages for the InUse Instance of the Protocol**

### 7 2.6.1 Protocol Initialization for the InUse Protocol Instance

8 Upon creation, the InUse instance of this protocol in the access terminal and access  
9 network shall perform the procedures specified in [1].

### 10 2.6.2 Hard Commit Procedures

11 The access terminal and the access network shall perform the procedures specified in [1]  
12 when directed by the InUse instance of the Session Control Protocol to execute the Hard  
13 Commit procedures.

### 14 2.6.3 Soft Commit Procedures

15 The access terminal and the access network shall perform the procedures specified in [1],  
16 in the order specified, when directed by the InUse instance of the Session Control Protocol  
17 to execute the Soft Commit procedures.

### 18 2.6.4 Procedures

19 BSP receives messages for transmission from multiple protocols. BSP shall add the BSP  
20 header to each message and forward it for transmission to RLP.

21 BSP receives messages from RLP. If BSP has received *RouteControl.RouteClosed* indication,  
22 then BSP shall ignore the messages received from RLP; otherwise, BSP shall route these  
23 messages to their associated instance of protocols according to the value of the Type field in  
24 the BSP header. ~~The BSP shall route the message to the instance of the protocol identified~~  
25 ~~by the Type field.~~

26 If BSP receives *RadioLinkNN.ReliablePacketDeliveryFailed* indication, where *NN* corresponds  
27 to an RLP instance associated with the BSP, then the BSP shall return a  
28 *ReliableMessageDeliveryFailed* indication.

29 If BSP receives *RadioLinkNN.ReliableStreamReset* indication, where *NN* corresponds to RLP  
30 instance associated with the BSP, then the BSP shall return a *ReliableSignalingStreamReset*  
31 indication.

### 32 2.6.5 Type Definitions

33 Protocol Type values assigned to protocols are specified in [1]. Protocol Type value for InUse  
34 and InConfiguration instance of this protocol is negotiated by Radio Link Protocol.

## 1 2.6.6 Messages

2 No messages are defined for the InUse instance of the protocol.

## 3 2.6.7 BSP Header

4 The BSP shall place the following header in front of every message that it sends:

5

Field	Length (bits)
Type	8 or 16

6 Type Protocol Type. This field shall be set to the Type value for the  
7 protocol (as defined in [1]) associated with the encapsulated  
8 message.

9 **2.7 Interface to Other Protocols**

## 10 2.7.1 Commands

11 This protocol does not issue any commands.

## 12 2.7.2 Indications

13 This protocol registers to receive the following indications:

- 14 • *RadioLinkNN.ReliablePacketDeliveryFailed*
- 15 • *RadioLinkNN.ReliableStreamReset*
- 16 • *RouteControl.RouteClosed*

17 **2.8 Configuration Attributes**

18 This protocol does not define any simple attributes or complex attributes.

19 **2.9 Non-Attribute Data**

20 This protocol does not define any non-attribute data.

21 **2.10 Session State Information**

22 The Session State Information record (see [1]) consists of parameter records. All  
23 configuration attributes and Non-attribute data are Session State Information records. This  
24 protocol does not define additional parameter records.

- 1 No text.

1 **3 BASIC INTER-ROUTE TUNNELING PROTOCOL**

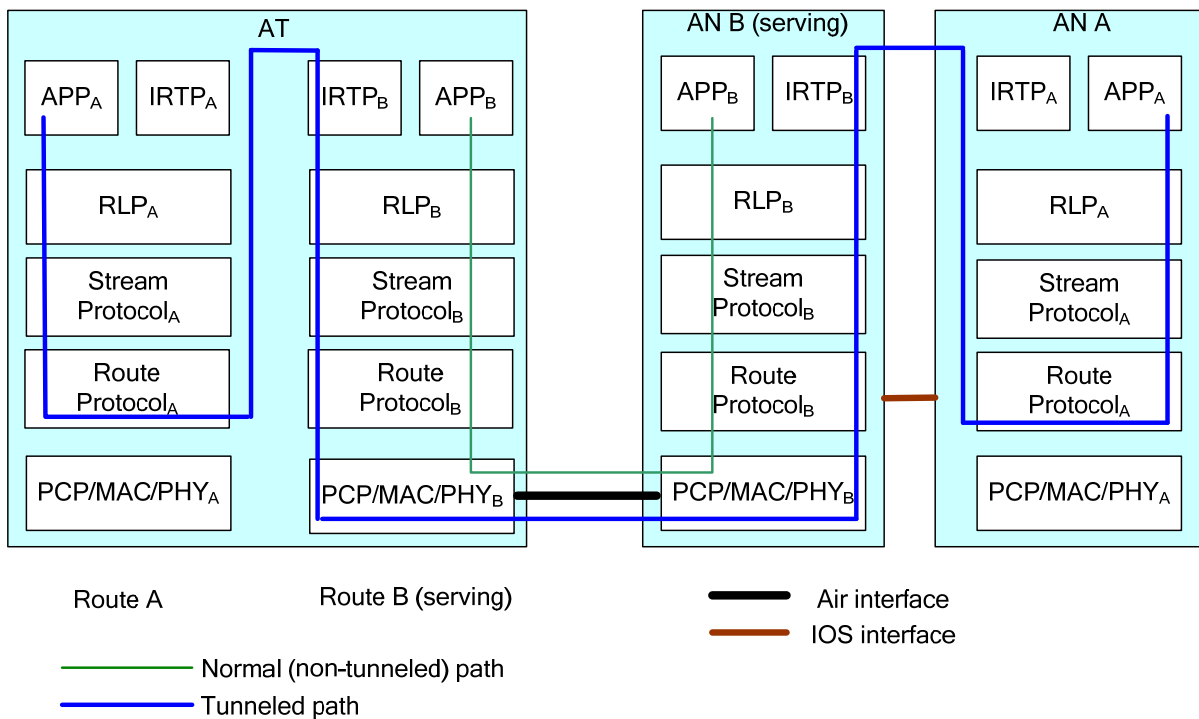
2 **3.1 Overview**

3 The Inter-Route Tunneling Protocol performs the following functions:

- 4 • Tunneling of data belonging to different Routes. The Inter-Route Tunneling Protocol  
 5 Header indicates the Route to which the payload belongs. The Inter-Route Tunneling  
 6 Protocol allows one Route to carry payload bound for another Route, including payload  
 7 bound for its Route.

8 A Route consists of an InUse protocol stack associated with an access network.

9 At the transmitter, the Inter-Route Tunneling Protocol receives packets for transmission  
 10 from the Route Protocol of another Route or from the Route Protocol of the same Route (see  
 11 Figure 3-1). The Inter-Route Tunneling Protocol adds an Inter-Route Tunneling Protocol  
 12 Header to the received packet to identify the destination Route and delivers this packet to  
 13 the Radio Link Protocol.



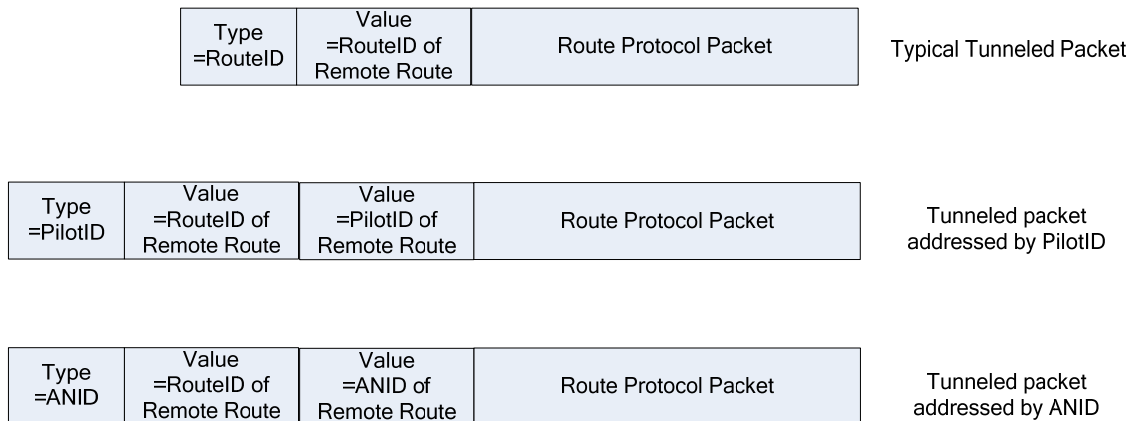
14

15 **Figure 3-1. Data Path for Normal and Tunneled Packets**

16 At the receiver, the Inter-Route Tunneling Protocol receives packets from the Radio Link  
 17 Protocol. The Inter-Route Tunneling Protocol removes the Inter-Route Tunneling Protocol  
 18 Header and delivers the packet to the Route Protocol of the corresponding Route.

19 Inter-Route Tunneling Protocol may receive packets from the Route Protocol of its Route for  
 20 further fragmentation at RLP.

1 Figure 3-2 illustrates various configurations of the Inter-Route Tunneling Protocol Header.



2

3

**Figure 3-2. Example Inter-Route Tunneling Protocol Headers**

## 4 **3.2 Primitives, Local Common Data, and Public Data**

### 5 3.2.1 Commands

6 This protocol does not define any commands.

### 7 3.2.2 Return Indications

8 This protocol does not return any indications.

### 9 3.2.3 Local Common Data

10 This protocol does not define any Local Common Data.

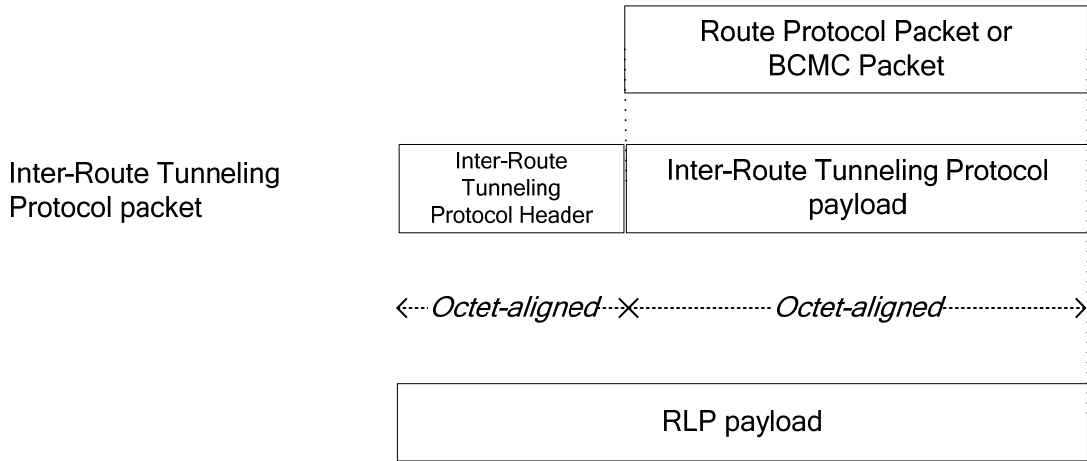
### 11 3.2.4 Public Data

12 This protocol shall make the following data public:

- 13 • All data defined as Static Attribute, Static Non-Attribute Data, and Local Common Data

## 14 **3.3 Protocol Data Unit**

15 The protocol data unit for this protocol is an Inter-Route Tunneling Protocol Packet. An  
 16 Inter-Route Tunneling Protocol Packet consists of an Inter-Route Tunneling Protocol  
 17 Payload and an Inter-Route Tunneling Protocol Header.



1

2

**Figure 3-3. Inter-Route Tunneling Protocol Packet**

3 **3.4 Procedures and Messages for the InConfiguration Instance of the Protocol**

4 3.4.1 Procedures

5 This protocol uses the services of the Session Control Protocol to perform negotiation of  
6 attribute values.

7 When configuring the Reservation *KKIRTPStreamRev* attribute for a Reservation *KK*, the  
8 access terminal shall not set it to a Stream that is not defined as a Inter-Route Tunneling  
9 Protocol stream by the Radio Link Protocol (see [4]).

10 3.4.1.1 Protocol Initialization

11 Upon creation, the InConfiguration instance of this protocol in the access terminal and the  
12 access network shall perform the procedures specified in [1].

13 3.4.2 Message Formats

14 This protocol does not define any messages.

15 **3.5 Procedures and Messages for the InUse Instance of the Protocol**

16 3.5.1 Procedures

17 3.5.1.1 Protocol Initialization

18 Upon creation, the InUse instance of this protocol in the access terminal and access  
19 network shall perform the procedures specified in [1].

20 3.5.1.2 Hard Commit Procedures

21 The access terminal and the access network shall perform the procedures specified in [1]  
22 when directed by the InUse instance of the Session Control Protocol to execute the Hard  
23 Commit procedures.

## 1 3.5.1.3 Soft Commit Procedures

2 The access terminal and the access network shall perform the procedures specified in [1]  
 3 when directed by the InUse instance of the Session Control Protocol to execute the Soft  
 4 Commit procedures.

## 5 3.5.1.4 Inter-Route Tunneling Protocol Header

6 The Inter-Route Tunneling Protocol Header has the following format:  
 7

Field	Length (bits)
HeaderType	1 or 4
If HeaderType= '0'	
RouteID	7
If HeaderType= '1000'	
RouteIDIncluded	1
RouteID	0 or 7
PilotID	10
If HeaderType= '1001'	
RouteIDIncluded	1
RouteID	0 or 7
Reserved1	3 or 4
ANIDmsb	64
If HeaderType= '1111'	
SpecialRouteID	4
Reserved2	0-7 (as needed)

8 HeaderType

The sender shall set this field as specified in Table 3-1 to  
 9 indicate the type of Inter-Route Tunneling Protocol Header.

10

**Table 3-1. HeaderType Values**

HeaderType (binary)	Type of Inter-Route Tunneling Protocol Header
0	RouteID Header
1000	PilotID Header
1001	ANID Header
1111	Special RouteID Header
Other values	Reserved



1 SpecialRouteID The sender shall set this field to the special Route Identifier,  
 2 as specified in Table 3-2, corresponding to the Route to which  
 3 this packet is destined.

4 **Table 3-2. SpecialRouteID Values**

<b>SpecialRouteID (binary)</b>	<b>SpecialRouteID usage</b>
0000	BCMC Route
0001-1111	Reserved

5 Reserved2 The sender shall include zero to seven bits to make this Inter-  
 6 Route Tunneling Protocol Header octet-aligned. The sender  
 7 shall set these bits to 0. The receiver shall ignore these bits.

### 8 3.5.1.5 Command Processing

#### 9 3.5.1.5.1 Activate

10 None.

#### 11 3.5.1.5.2 Deactivate

12 None.

### 13 3.5.1.6 Inter-Route Tunneling Protocol Stream Mapping Procedures

14 If the State public data of the Route Control Protocol is WaitingToOpen, the access terminal  
 15 shall map all packets to Stream 0x04; otherwise, the access terminal shall perform the  
 16 following:

- 17 • The access terminal shall map all packets not associated with a Reservation*KK* (i.e.,  
 18 packets generated by applications other than IP or ROHC) to Stream 0x04
- 19 • The access terminal should map all packets associated with a Reservation*KK* to the  
 20 Stream as specified by Reservation*KK*IRTPStreamRev

21 If the State public data of the Route Control Protocol is WaitingToOpen, the access network  
 22 shall map all packets to Stream 0x04.

### 23 3.5.1.7 Transmission Procedures

#### 24 3.5.1.7.1 Access Terminal Procedure

25 If the Inter-Route Tunneling Protocol receives a Route Protocol Packet from the Route  
 26 Protocol of a Route, it shall perform the following:

- 27 • The access terminal shall satisfy the following requirements in setting the fields of the  
 28 Inter-Route Tunneling Protocol header:

- 1 ~~If the State public data of the Route Control Protocol is WaitingToOpen, the access~~  
2 ~~terminal shall not use the following HeaderTypes:~~  
3 ~~+HeaderType of '1000'~~  
4 ~~+HeaderType of '1001' unless it is in response to a RouteCreate message~~  
5 ~~– Otherwise~~If the State public data of the Route Control Protocol is Open, the access  
6 terminal shall not use HeaderType of '0' for a destination Route if the access  
7 terminal has not received the acknowledgement for the RouteMap information sent  
8 to this access network containing the ANID to RouteID mapping for this destination  
9 Route.
- 10 • The Inter-Route Tunneling Protocol shall add an Inter-Route Tunneling Protocol Header  
11 to identify the destination Route.
  - 12 • The Inter-Route Tunneling Protocol shall deliver the Inter-Route Tunneling Protocol  
13 Packet to the Radio Link Protocol of its Route.

#### 14 3.5.1.7.2 Access Network Procedures

15 If the Inter-Route Tunneling Protocol receives a Route Protocol Packet from the Route  
16 Protocol of a Route, it shall perform the following:

- 17 • The access network shall satisfy the following requirements in setting the fields of the  
18 Inter-Route Tunneling Protocol header:
  - 19 – The access network shall not use HeaderType '1001' (ANID) in the Inter-Route  
20 Tunneling Protocol Header ~~until the RouteMap information is received.~~
- 21 • The Inter-Route Tunneling Protocol shall add an Inter-Route Tunneling Protocol Header  
22 to identify the destination Route.
- 23 • The Inter-Route Tunneling Protocol shall deliver the Inter-Route Tunneling Protocol  
24 Packet to the Radio Link Protocol of its Route.

#### 25 3.5.1.8 Reception Procedures

##### 26 3.5.1.8.1 Access Terminal Procedures

27 If the Inter-Route Tunneling Protocol receives an Inter-Route Tunneling Protocol Packet  
28 from the Radio Link Protocol of its Route, it shall perform the following:

- 29 • The Inter-Route Tunneling Protocol shall remove the Inter-Route Tunneling Protocol  
30 Header to produce an Inter-Route Tunneling Protocol Payload.
- 31 • The Inter-Route Tunneling Protocol shall perform the following:
  - 32 – If the HeaderType field of the header is set to '0', the Inter-Route Tunneling Protocol  
33 shall deliver the Inter-Route Tunneling Protocol Payload to the Route Protocol of the  
34 Route identified in the RouteID field of the header.

- 1       – If the HeaderType field of the header is set to ‘1000’, the Inter-Route Tunneling  
2       Protocol shall deliver the Inter-Route Tunneling Protocol Payload to the Route  
3       Protocol associated with the pilot identified by the PilotID field of the header, along  
4       with the PilotID value<sup>3</sup>.

5       ~~If the HeaderType field of the header is set to ‘1001’, the Inter-Route Tunneling  
6       Protocol shall deliver the Inter-Route Tunneling Protocol Payload to the Route  
7       Protocol associated with the Route associated with the received ANIDmsb.~~

#### 8   3.5.1.8.2 Access Network Procedures

9   If the Inter-Route Tunneling Protocol receives an Inter-Route Tunneling Protocol Packet  
10 from the Radio Link Protocol of its Route, it shall perform the following:

- 11   • The Inter-Route Tunneling Protocol shall remove the Inter-Route Tunneling Protocol  
12   Header to produce an Inter-Route Tunneling Protocol Payload.
- 13   • If the HeaderType field of the header is set to ‘0’, the access network shall perform the  
14   following:
  - 15   – If the access network does not recognize the RouteID field of the header:
    - 16   + If the State public data of the Route Control Protocol is WaitingToOpen, the  
17   Inter-Route Tunneling Protocol shall deliver a copy of the Inter-Route Tunneling  
18   Protocol Payload to the Route Protocol of the SessionAnchor Route.
    - 19   + Otherwise, the Inter-Route Tunneling Protocol shall issue a  
20   *RouteControl.QueryRouteMap* command.
  - 21   – Otherwise, the Inter-Route Tunneling Protocol shall deliver a copy of the Inter-Route  
22   Tunneling Protocol Payload to the Route Protocol of the Route identified in the  
23   RouteID field of the header.
- 24   • If the HeaderType field of the header is set to ‘1000’, the Inter-Route Tunneling Protocol  
25   shall deliver the Inter-Route Tunneling Protocol Payload to the Route Protocol of the  
26   access network to which the sector identified in the PilotID field of the header belongs  
27   to.
- 28   • If the HeaderType field of the header is set to ‘1001’, the Inter-Route Tunneling Protocol  
29   shall deliver the Inter-Route Tunneling Protocol Payload to the Route Protocol of the  
30   access network corresponding to the received access network identifier.

#### 31   3.5.2 Type Definitions

32   Protocol Type values assigned to protocols are specified in [1]. Protocol Type value for InUse  
33   and InConfiguration instance of this protocol is negotiated by Radio Link Protocol.

#### 34   3.5.3 Message Formats

35   This protocol does not define any messages.

---

<sup>3</sup> Since separate RLP queue ~~is may be~~ maintained for each sector of a Route, the PilotID information is necessary to deliver this packet to the correct RLP queue.

### 1 **3.6 Interface to Other Protocols**

#### 2 3.6.1 Commands Sent

3 This protocol issues the following commands:

- 4 • *RouteControl.QueryRouteMap*

#### 5 3.6.2 Indications

6 This protocol does not register to receive any indications.

### 7 **3.7 Configuration Attributes**

#### 8 3.7.1 Simple Attributes

9 The negotiable simple attributes for this protocol is listed in Table 3-3. The access terminal  
10 and the access network shall use as defaults the values in Table 3-3 typed in ***bold italics***.

11 **Table 3-3. Configurable Values**

Attribute ID	Attribute	Commit/ Scope	Values	Meaning
0x01 <i>KK</i> <i>KK</i> is the two-digit hexadecimal Reservation Label.	Reservation <i>KKIRTP</i> StreamRev <i>KK</i> is the two-digit hexadecimal Reverse Link ReservationLabel, where hexadecimal digits A through F are specified in upper case letters.	Soft/ Dynam ic	<b>0x06</b>	Stream 0x06 is associated with this Reverse ReservationLabel.
			0x04-0x05, 0x07-0x1e	Stream associated with this Reverse ReservationLabel.
			All other values	Reserved

#### 12 3.7.2 Complex Attributes

13 This protocol does not define any complex attributes.

### 14 **3.8 Non-Attribute Data**

15 This protocol does not define any non-attribute data.

### 16 **3.9 Session State Information**

17 The Session State Information record (see [1]) consists of parameter records. All  
18 configuration attributes and non-attribute data are Session State Information records. This  
19 protocol does not define additional parameter records.

- 1 No text.

## 1 **4 BASIC ROHC SUPPORT PROTOCOL**

### 2 **4.1 Overview**

3 The ROHC Support Protocol performs the following functions:

- 4 • Negotiate ROHC attributes
- 5 • Sending and Receiving ROHC protocol packets

### 6 **4.2 Primitives, Local Common Data, and Public Data**

#### 7 4.2.1 Commands

8 This protocol does not define any commands.

#### 9 4.2.2 Return Indications

10 This protocol does not return any indications.

#### 11 4.2.3 Local Common Data

12 This protocol does not define any Local Common Data.

#### 13 4.2.4 Public Data

14 This protocol shall make the following data public:

- 15 • All data defined as Static Attribute, Static Non-Attribute Data, and Local Common Data

### 16 **4.3 Protocol Data Unit**

17 The protocol data unit for this protocol is a ROHC Packet. This protocol does not add any  
18 header.

### 19 **4.4 Protocol Initialization**

### 20 **4.5 Procedures and Messages for the InConfiguration Instance of the Protocol**

#### 21 4.5.1 Protocol Initialization for the InConfiguration Protocol Instance

22 Upon creation, the InConfiguration instance of this protocol in the access terminal and the  
23 access network shall perform the procedures specified in [1].

#### 24 4.5.2 Procedures

25 This protocol uses the services of the Session Control Protocol to perform negotiation of  
26 attribute values.

#### 27 4.5.3 Message Formats

28 This protocol does not define any messages.

## 1 **4.6 Procedures and Messages for the InUse Instance of the Protocol**

### 2 4.6.1 Procedures

#### 3 4.6.1.1 Protocol Initialization for the InUse Protocol Instance

4 Upon creation, the InUse instance of this protocol in the access terminal and access  
5 network shall perform the procedures specified in [1].

#### 6 4.6.1.2 Hard Commit Procedures

7 The access terminal and the access network shall perform the procedures specified in [1]  
8 when directed by the InUse instance of the Session Control Protocol to execute the Hard  
9 Commit procedures.

#### 10 4.6.1.3 Soft Commit Procedures

11 The access terminal and the access network shall perform the procedures specified in [1]  
12 when directed by the InUse instance of the Session Control Protocol to execute the Soft  
13 Commit procedures.

#### 14 4.6.1.4 ROHC Support Protocol Header

15 The ROHC Support Protocol does not add any header.

#### 16 4.6.1.5 Command Processing

17 This protocol does not define any commands.

#### 18 4.6.1.6 Common Procedures

19 Access network and access terminal send and receive packets as defined in [12].

##### 20 4.6.1.6.1 Use of Multiple ROHC Versions

21 The ROHC profile is a 16-bit integer. The 8 LSBs distinguish the profile (e.g., RTP, UDP etc)  
22 and the 8 MSBs distinguish the ROHC version. In ROHC packets, the profile identifier is  
23 abbreviated as an 8-bit field, carrying only the 8 LSBs. Thus, if two or more profiles with  
24 the same 8 LSBs are supported, ROHC needs a way to uniquely identify the version of the  
25 profile being used.

26 The access terminal shall use the 16-bit representation of the profiles carried in the Profiles  
27 field of ROHCParametersFwd to identify the ROHC version that the 8-bit profile identifier  
28 carried in the ROHC packet refers to. The access network shall use the 16-bit  
29 representation of the profiles carried in the Profiles field of ROHCParametersRev to identify  
30 the ROHC version that the 8-bit profile identifier carried in the ROHC packet refers to.

### 31 4.6.2 Type Definitions

32 Protocol Type values assigned to protocols are specified in [1]. Protocol Type value for InUse  
33 and InConfiguration instance of this protocol is negotiated by Radio Link Protocol.

## 1 4.6.3 Message Formats

2 This protocol does not define any messages.

3 **4.7 Interface to Other Protocols**

## 4 4.7.1 Commands Sent

5 This protocol does not issue any commands.

## 6 4.7.2 Indications

7 This protocol does not register to receive any indications.

8 **4.8 Configuration Attributes**

## 9 4.8.1 Simple Attributes

10 This protocol does not define any simple attributes.

## 11 4.8.2 Complex Attributes

12 This protocol defines following complex attributes.

## 13 4.8.2.1 ROHCParametersFwd Attribute

14 The sender shall set AttributeID field to 0x8000.

15

Field	Length (bits)	Default
MaxCID	16	0
LargeCIDs	1	0
FeedbackForIncluded	1	N/A
FeedbackFor	0 or 5	N/A
MRRU	16	0
ProfileCount	8	4

ProfileCount occurrences of the following field:

Profile	16	0x0000 (uncompressed), 0x0001 (RTP), 0x0002 (UDP), 0x0003 (ESP)
---------	----	---

Reserved	0 - 7 (as needed)	N/A
----------	----------------------	-----

1	MaxCID	The sender shall set this field to the MAX_CID parameter (See [12]) for this ROHC Channel. The sender shall not set this field to a value greater than MaxSupportedMaxCID.
2		
3		
4	LargeCIDs	If the LARGE_CIDS parameter (See [12]) for this ROHC Channel is false, then the sender shall set this field to '0'. Otherwise, the sender shall set this field to '1'. The sender shall not set this field to '1' if LargeCIDsSupported is not set to '1'.
5		
6		
7		
8		
9	FeedbackForIncluded	If ROHC feedback associated with another Stream (ROHC channel) is sent on this Stream (ROHC channel), then this field shall be set to '1'. Otherwise, this field shall be set to '0'.
10		
11		
12	FeedbackFor	If FeedbackForIncluded is set to '0', then the sender shall omit this field. Otherwise, the sender shall set this field to the Stream number (ROHC channel) to which ROHC feedback sent on this Stream (ROHC channel) refers.
13		
14		
15		
16	MRRU	The sender shall set this field to the MRRU parameter for this ROHC channel. The sender shall not set this field to a value larger than MaxSupportedMRRU.
17		
18		
19	ProfileCount	The sender shall set this field to the number of ROHC profiles supported by the decompressor.
20		
21	Profile	The sender shall set this field to the ROHC profile supported by the decompressor according to [13]. The sender shall not set this field to a value that is not included in the list of supported Profiles. The sender shall not set two or more Profile values with same value of 8 LSBs.
22		
23		
24		
25		
26	Reserved	The sender shall add reserved bits to make the length of the entire record an integer number of octets. The sender shall set these bits to '0'. The receiver shall ignore this field.
27		
28		
29		

<b>Commit</b>	Soft	<b>Scope</b>	Dynamic
---------------	------	--------------	---------

#### 30 4.8.2.2 ROHCParametersRev Attribute

31 The sender shall set AttributeID field to 0x8001.

32

Field	Length (bits)	Default
MaxCID	16	0
LargeCIDs	1	0
FeedbackForIncluded	1	N/A
FeedbackFor	0 or 5	N/A
MRRU	16	0
TimerBasedCompression	1	1
ProfileCount	8	4

ProfileCount occurrences of the following field:

Profile	16	0x0000 (uncompressed), 0x0001 (RTP), 0x0002 (UDP), 0x0003 (ESP)
Reserved	0 - 7 (as needed)	N/A

- 1 MaxCID The sender shall set this field to the MAX\_CID parameter (See  
2 [12]) for this ROHC Channel. The sender shall not set this  
3 field to a value greater than MaxSupportedMaxCID.
- 4 LargeCIDs If the LARGE\_CIDS parameter (See [12]) for this ROHC  
5 Channel is false, then the sender shall set this field to '0'.  
6 Otherwise, the sender shall set this field to '1'. If  
7 LargeCIDsSupported is '0', then the sender shall not set this  
8 field to '1'.
- 9 FeedbackForIncluded If ROHC feedback associated with another Stream (ROHC  
10 channel) is sent on this Stream (ROHC channel), then this  
11 field shall be set to '1'. Otherwise, this field shall be set to '0'.
- 12 FeedbackFor If FeedbackForIncluded is set to '0', then the sender shall  
13 omit this field. Otherwise, the sender shall set this field to the  
14 Stream number (ROHC channel) to which ROHC feedback  
15 sent on this Stream (ROHC channel) refers.
- 16 MRRU The sender shall set this field to the MRRU parameter for this  
17 ROHC channel. The sender shall not set this field to a value  
18 greater than MaxSupportedMRRU.

1	TimerBasedCompression	The sender shall set this field to '0' if timer based compression according to [12] is not enabled for this ROHC channel. The sender shall set this field to '1' if timer based compression according to [12] is enabled for this ROHC channel. If TimerBasedCompressionSupported is set to '0', then the sender shall not set this field to '1'.
2		
3		
4		
5		
6		
7	ProfileCount	The sender shall set this field to the number of ROHC profiles supported by the decompressor.
8		
9	Profile	The sender shall set this field to the ROHC profile supported by the decompressor according to [13]. The sender shall not set this field to a value that is not included in the list of supported profiles. The sender shall not set two or more Profile values with same value of 8 LSBs.
10		
11		
12		
13		
14	Reserved	The sender shall add reserved bits to make the length of the entire record an integer number of octets. The sender shall set these bits to '0'. The receiver shall ignore this field.
15		
16		
17		

<b>Commit</b>	Soft
---------------	------

<b>Scope</b>	Dynamic
--------------	---------

#### 18 **4.9 Non-Attribute Data**

19 This protocol does not define any non-attribute data.

#### 20 **4.10 Session State Information**

21 The Session State Information record (see [1]) consists of parameter records. All  
 22 configuration attributes and Non-attribute data are Session State Information records. This  
 23 protocol does not define additional parameter records.

## 1 **5 BASIC EAP SUPPORT PROTOCOL**

### 2 **5.1 Overview**

3 The EAP Support Protocol performs the following functions:

- 4 • Sending and Receiving EAP protocol packets
- 5 • Negotiate if the access terminal would initiate EAP Fast Reauthentication when Route  
6 Open is initiated
- 7 • Provide control messages for EAP

### 8 **5.2 Primitives, Local Common Data, and Public Data**

#### 9 5.2.1 Commands

10 This protocol does not define any commands.

#### 11 5.2.2 Return Indications

12 This protocol does not return any indications.

#### 13 5.2.3 Local Common Data

14 This protocol does not define any Local Common Data.

#### 15 5.2.4 Public Data

16 This protocol shall make the following data public:

- 17 • All data defined as Static Attribute, Static Non-Attribute Data, and Local Common Data

### 18 **5.3 Protocol Data Unit**

19 The protocol data unit for this protocol is an EAP Packet. This protocol does not add any  
20 header.

### 21 **5.4 Protocol Initialization**

### 22 **5.5 Procedures and Messages for the InConfiguration Instance of the Protocol**

#### 23 5.5.1 Protocol Initialization for the InConfiguration Protocol Instance

24 Upon creation, the InConfiguration instance of this protocol in the access terminal and the  
25 access network shall perform the procedures specified in [1].

#### 26 5.5.2 Procedures

27 This protocol uses the services of the Session Control Protocol to perform negotiation of  
28 attribute values.

1 5.5.3 Message Formats

2 This protocol does not define any messages.

3 **5.6 Procedures and Messages for the InUse Instance of the Protocol**

4 5.6.1 Procedures

5 5.6.1.1 Protocol Initialization for the InUse Protocol Instance

6 Upon creation, the InUse instance of this protocol in the access terminal and access  
7 network shall perform the procedures specified in [1].

8 5.6.1.2 Hard Commit Procedures

9 The access terminal and the access network shall perform the procedures specified in [1]  
10 when directed by the InUse instance of the Session Control Protocol to execute the Hard  
11 Commit procedures.

12 5.6.1.3 Soft Commit Procedures

13 The access terminal and the access network shall perform the procedures specified in [1]  
14 when directed by the InUse instance of the Session Control Protocol to execute the Soft  
15 Commit procedures.

16 5.6.1.4 EAP Support Protocol Header

17 The EAP Support Protocol does not add any header.

18 5.6.1.5 Command Processing

19 This protocol does not define any commands.

20 5.6.1.6 Common Procedures

21 Access network and access terminal send and receive EAP packets as defined in [14].

22 5.6.1.7 Access Terminal Procedures

23 The access terminal shall initiate EAP Fast Reauthentication procedure when Route Open  
24 is initiated, if and only if any of the following conditions is true:

- 25 • InitiateEAPFastReauthentication attribute of this protocol in the Session Anchor Route  
26 | is set to 0x0000000x01.

27 The access terminal may send InitiateEAP message to request access network to initiate  
28 EAP procedure as specified in [14].

29 If access terminal receives InitiateEAPFastReauthentication message and the  
30 EAPFastReauthenticationSupported field of ATSupportedApplicationProtocolIDs public  
31 data of the Session Control Protocol is set to '1', the access terminal shall perform one of  
32 the following:

- 33 • Initiate EAP Fast Reauthentication procedure.

- 1 • Send InitiateEAP message.

2 5.6.1.8 Access Network Procedures

3 If the EAPFastReauthenticationSupported field of ATSupportedApplicationProtocolIDs  
 4 public data of the Session Control Protocol is set to '1', then the access network may send  
 5 InitiateEAPFastReauthentication message to request access terminal to initiate EAP Fast  
 6 Reauthentication procedure.

7 If access network receives InitiateEAP message, the access network shall initiate EAP  
 8 procedure as specified in [14].

9 5.6.2 Type Definitions

10 Protocol Type values assigned to protocols are specified in [1]. Protocol Type value for InUse  
 11 and InConfiguration instance of this protocol is negotiated by Radio Link Protocol.

12 5.6.3 Message Formats

13 5.6.3.1 InitiateEAPFastReauthentication

14 The sender sends this message to request the access terminal to initiate EAP Fast  
 15 Reauthentication procedures.

16

Field	Length (bits)
MessageID	8

17 MessageID

The sender shall set this field to 0x00.

18

<b>Channels</b>	FTC    RTC	<b>RLP</b>	Reliable
<b>Addressing</b>	unicast	<b>AuthTag</b>	Required when key is available

19 5.6.3.2 InitiateEAP

20 The access terminal sends this message to request the access network to initiate EAP  
 21 procedures.

22

Field	Length (bits)
MessageID	8

23 MessageID

The sender shall set this field to 0x01.

24

<b>Channels</b>	RTC	<b>RLP</b>	Reliable
<b>Addressing</b>	unicast	<b>AuthTag</b>	Required when key is available

## 1 5.7 Interface to Other Protocols

### 2 5.7.1 Commands Sent

3 This protocol does not issue any commands.

### 4 5.7.2 Indications

5 This protocol does not register to receive any indications.

## 6 5.8 Configuration Attributes

### 7 5.8.1 Simple Attributes

8 The negotiable simple attributes for this protocol is listed in Table 5-1. The access terminal  
9 and the access network shall use as defaults the values in Table 5-1 typed in ***bold italics***.

10 **Table 5-1. Configurable Values**

Attribute ID	Attribute	Commit/ Scope	Values	Meaning
0x0000	InitiateEAPFastReauthenticati on	Soft/ Dynami c	<del>0x0000</del> <del>000x01</del>	Access terminal to initiate EAP Fast Reauthentication when Route Open is initiated.
			<del>0x111</del> <del>1110x</del> <del>00</del>	Access terminal to not initiate EAP Fast Reauthentication when Route Open is initiated.
			All other values	Reserved

### 11 5.8.2 Complex Attributes

12 This protocol does not define any complex attributes.

## 13 5.9 Non-Attribute Data

14 This protocol does not define any non-attribute data.

## 15 5.10 Session State Information

16 The Session State Information record (see [1]) consists of parameter records. All  
17 configuration attributes and Non-attribute data are Session State Information records. This  
18 protocol does not define additional parameter records.