International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

H.235.4

(09/2005)

SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS

Infrastructure of audiovisual services – Systems aspects

H.323 security: Direct and selective routed call security

ITU-T Recommendation H.235.4

ITU-T H-SERIES RECOMMENDATIONS

AUDIOVISUAL AND MULTIMEDIA SYSTEMS

CHARACTERISTICS OF VISUAL TELEPHONE SYSTEMS	H.100–H.199
INFRASTRUCTURE OF AUDIOVISUAL SERVICES	
General	H.200–H.219
Transmission multiplexing and synchronization	H.220–H.229
Systems aspects	**H.230–H.239**
Communication procedures	H.240–H.259
Coding of moving video	H.260–H.279
Related systems aspects	H.280–H.299
Systems and terminal equipment for audiovisual services	H.300–H.349
Directory services architecture for audiovisual and multimedia services	H.350–H.359
Quality of service architecture for audiovisual and multimedia services	H.360–H.369
Supplementary services for multimedia	H.450–H.499
MOBILITY AND COLLABORATION PROCEDURES	
Overview of Mobility and Collaboration, definitions, protocols and procedures	H.500–H.509
Mobility for H-Series multimedia systems and services	H.510–H.519
Mobile multimedia collaboration applications and services	H.520–H.529
Security for mobile multimedia systems and services	H.530–H.539
Security for mobile multimedia collaboration applications and services	H.540–H.549
Mobility interworking procedures	H.550–H.559
Mobile multimedia collaboration inter-working procedures	H.560–H.569
BROADBAND AND TRIPLE-PLAY MULTIMEDIA SERVICES	
Broadband multimedia services over VDSL	H.610–H.619

For further details, please refer to the list of ITU-T Recommendations.

ITU-T Recommendation H.235.4

H.323 security: Direct and selective routed call security

Summary

The purpose of this Recommendation is to provide recommendations of security procedures for using direct-routed call signalling in conjunction with H.235.1 and H.235.3 security profiles. This security profile is offered as an option and may complement the security profiles in ITU-T Recs H.235.1 and H.235.3. It also provides implementation details for clause 8.4/H.235.0 using symmetric key management techniques.

In earlier versions of the H.235 subseries, this profile was contained in Annex I/H.235. Appendices IV, V, VI to H.235.0 show the complete clause, figure, and table mapping between H.235 versions 3 and 4.

Source

ITU-T Recommendation H.235.4 was approved on 13 September 2005 by ITU-T Study Group 16 (2005-2008) under the ITU-T Recommendation A.8 procedure.

Keywords

Authentication, direct-routed call security, encryption, integrity, key management, multimedia security, security profile, selective routed call security.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure e.g. interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

CONTENTS

ITU-T Recommendation H.235.4

H.323 security: Direct and selective routed call security

1 Scope

The purpose of this Recommendation is to provide recommendations of security procedures for using direct-routed and selective routed call signalling in conjunction with H.235.1 and H.235.3 security profiles.

This security profile is offered as an option and may complement the H.235.1 or H.235.3 security profiles. It also provides implementation details for clause 8.4/H.235.0 using symmetric key management techniques.

2 References

2.1 Normative references

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

– ITU-T Recommendation H.225.0 (2003), *Call signalling protocols and media stream packetization for packet-based multimedia communication systems*.

– ITU-T Recommendation H.235 (2003), *Security and encryption for H-series (H.323 and other H.245-based) multimedia terminals*, Corrigendum 1 (2005), plus Erratum 1 (2005).

– ITU-T Recommendation H.235.0 (2005), *H.323 security: Framework for security in H-series (H.323 and other H.245-based) multimedia systems*.

– ITU-T Recommendation H.235.1 (2005), *H.323 security: Baseline security profile*.

– ITU-T Recommendation H.235.3 (2005), *H.323 security: Hybrid security profile*.

– ITU-T Recommendation H.235.6 (2005), *H.323 security: Voice encryption profile with native H.235/H.245 key management.*

– ITU-T Recommendation H.323 (2003), *Packet-based multimedia communications systems*.

– ITU-T Recommendation X.800 (1991), *Security architecture for Open Systems Interconnection for CCITT applications*.

ISO/IEC 7498-2:1989, *Information processing systems – Open Systems Interconnection – Basic Reference model – Part 2: Security Architecture*.

– ISO/IEC 10118-3:2004, *Information technology – Security techniques – Hash functions – Part 3: Dedicated hash-functions*.

2.2 Informative references

– ITU-T Recommendation H.235.2 (2005), *H.323 security: Signature security profile*.

– IETF RFC 4120 (2005), *The Kerberos Network Authentication Service (V5).*

3 Terms and definitions

For the purposes of this Recommendation the definitions given in clause 3 of ITU-T Recs H.323, H.225.0, H.235.0 and X.800 | ISO 7498-2 apply.

4 Symbols and abbreviations

This Recommendation uses the following abbreviations:

CT	ClearToken
DH	Diffie-Hellman
DRC	Direct-Routed Call
EK_{AG}	The encryption key shared between EP A and GK G
EK_{BH}	The encryption key shared between EP B and GK H
EK_{GH}	The encryption key shared between GK G and GK H
$ENC_{K;\,S,\,IV}(M)$	EOFB Encryption of *M* using secret key *K* and secret salting key *S* and initial vector *IV*
EPID	Endpoint Identifier
GK	Gatekeeper
GKID	Gatekeeper Identifier
g^x, g^y	Diffie-Hellman half-key of GK G, GK H
K_{AB}	The encryption key shared between EP A and EP B
K_{AG}	Shared secret (H.235.1, H.235.3) between EP A and GK G
K_{BH}	Shared secret (H.235.1, H.235.3) between EP B and GK H
K_{GH}	Secret, secret (H.235.1, H.235.3) between GK G and GK H
KS_{AG}	Secret, shared salting key between EP A and GK G
KS_{BH}	Secret, shared salting key between EP B and GK H
KS_{GH}	Secret, shared salting key between GK G and GK H
PRF	Pseudo-Random Function

5 Conventions

In this Recommendation the following conventions are used:

– "shall" indicates a mandatory requirement.

– "should" indicates a suggested but optional course of action.

– "may" indicates an optional course of action rather than a recommendation that something take place.

The object identifiers are referenced through a symbolic reference in the text (e.g., "I11"), clause 14 lists the actual numeric values for the symbolic object identifiers, see also clause 5/H.235.0.

6 Introduction

H.323 is often deployed using the gatekeeper-routed model (for example, to take advantage of better billing functionalities). The widespread use of gatekeeper-routed call models is also the reason why different security profiles, focused exactly on this call model, are defined within ITU-T Rec. H.235.0 (such as H.235.1, H.235.2, H.235.3).

However, with the need to support an increasing number of parallel channels, the direct-routed call model with a gatekeeper could yield better performance and scalability properties. The advantage of this mode is the utilization of a gatekeeper for registration, admission, address resolution, and bandwidth control, while performing the call establishment directly between the end points in an end-to-end fashion.

This Recommendation describes the enhancements for the H.235.1 baseline and for H.235.3 hybrid security profiles to support direct-routed calls with gatekeeper(s).

7 Overview

The H.235.1 baseline, as well as the H.235.3 hybrid security profiles, apply a shared secret (after the first handshake) to assure message authentication and/or integrity in a hop-by-hop fashion using the gatekeeper as a trusted intermediate host. Using the direct-routed call model, a shared secret between two endpoints cannot be assumed. It is also not practical to use a pre-established shared secret to secure the communication since, in this case, all endpoints would have to know in advance which other endpoint will be called.

ITU-T Rec. H.235.4 addresses the scenario shown in Figure 1, where endpoints are attached to a gatekeeper and deploy direct-routed call signalling. The scenario assumes an unsecured IP network in the gatekeeper zone.

It is assumed that each endpoint has a communication relation and a security association with its gatekeeper, and that each endpoint has registered securely with the gatekeeper using either the baseline or the hybrid security profile.

Hence, the gatekeeper of the initiating endpoint (DRC1) or the gatekeeper of the terminating endpoint (DRC2) is able to provide a shared secret for the directly communicating endpoints using a Kerberos-like approach (see RFC 4120).

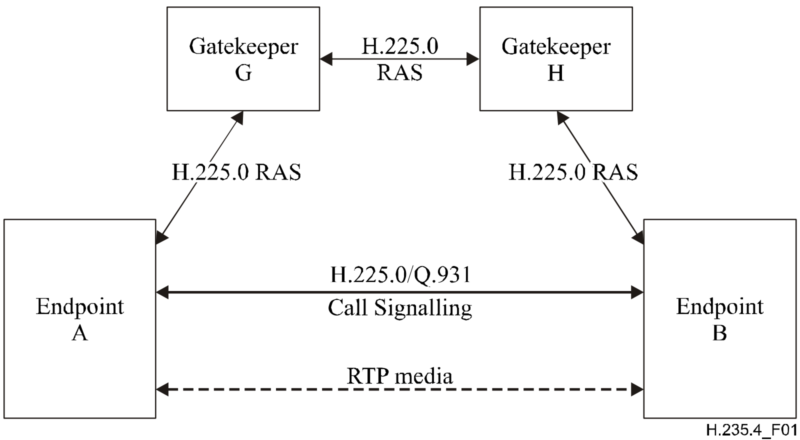

Figure 1/H.235.4 – Direct-routed call scenario

This Recommendation features two procedures, DRC1 and DRC2, for different environments.

Procedure DRC1 (see clause 9) is applicable in corporate environments where the gatekeepers are situated within different (local) sites but where the sites adhere to a common corporate security policy. In such an environment it is assumed acceptable that the originating gatekeeper G determines the effective security policy for a call to be established; thus the originating gatekeeper G selects and chooses the applied security parameters. The terminating gatekeeper H will accept the chosen security parameters.

Procedures DRC2 (see clause 10) and DRC3 (clause 11) are applicable in interdomain environments where the gatekeepers are situated within different administrative domains where each domain may employ a different security policy.

Procedure DRC2 is applicable in cases where the calling endpoint or the gatekeepers do not support the Diffie-Hellman algorithm. In such an environment it is assumed acceptable that the terminating gatekeeper H determines the effective security policy for a call to be established; thus the terminating gatekeeper H selects and chooses the applied security parameters. The originating gatekeeper G will accept the chosen security parameters.

Procedure DRC3 is applicable in cases where the calling endpoint does not support the Diffie-Hellman algorithm while the Gatekeepers in the calling and called domain both support the Diffie-Hellman algorithm.

At the beginning of call registration, the procedures provide signalling means to negotiate which of DRC1, DRC2 or DRC3 is to be applied.

8 Limitations

This Recommendation does not address direct-routed scenarios without any gatekeeper. This remains for further study.

9 Procedure DRC1 (corporate environment)

The procedure described in this clause is applicable in corporate environments where the gatekeepers are situated within different (local) sites but where the sites adhere to a common corporate security policy. In such an environment, it is assumed acceptable that the originating gatekeeper G determines the effective security policy for a call to be established; thus the originating gatekeeper selects and chooses the applied security parameters. The terminating gatekeeper H will accept the chosen security parameters.

9.1 GRQ/RRQ phase

Endpoints capable of supporting this security profile shall indicate this fact during **GRQ** and/or **RRQ** by including a separate ClearToken with **tokenOID** set to "I10"; any other fields in that ClearToken should not be used. The H.235.4-capable gatekeeper that is willing to provide this functionality shall reply with **GCF** or **RCF** with a separate ClearToken included with **tokenOID** set to "I10" and all other fields in the ClearToken unused.

9.2 ARQ phase

Before an endpoint A starts sending call signalling messages to another endpoint B directly, the endpoint A or B shall apply for admission at the gatekeeper G or H using **ARQ**. Endpoint A shall include within **ARQ** a separate ClearToken with **tokenOID** set to "I10" and all other fields in the ClearToken unused.

9.3 LRQ phase

This procedure covers the case of both a single, common gatekeeper to the endpoints and the case of multiple, chained gatekeepers. In the case of multiple involved gatekeepers, gatekeeper G, in which zone the call originates, should locate gatekeeper H using the (multicast) **LRQ** mechanism as described by ITU-T Rec. H.323 clause 8.1.6, "Optional called endpoint signalling". The communication between two gatekeepers shall be secured according to H.235.1. For this, it is assumed that a common shared secret K_{GH} is available. Since **LRQ** among gatekeepers is typically a multicast message, the shared secret K_{GH} typically cannot be a pair-wise shared secret but is assumed to be actually a group-based shared secret within the potential cloud of gatekeepers.

NOTE – This assumption limits scalability in the general case, and does not allow source authentication. However, it is believed that in corporate networks, with a limited, small number of well-known gatekeepers, such constraint and security limitations are still acceptable. Securing inter-gatekeeper multicast communication using digital signatures could overcome those limitations: however, this remains for further study.

If the **LRQ** mechanism is used to locate the far-end gatekeeper, then **LRQ** shall convey a separate ClearToken with **tokenOID** set to "I10"; any other fields in that ClearToken should not be used. For the multicast case, the **generalID** in the ClearToken of **LRQ** shall not be used. Intergatekeeper communication using H.501 and/or H.510 remains for further study.

9.4 LCF phase

EK_{BH} denotes the encryption key and KS_{BH} denotes the salting key that are shared between endpoint B and gatekeeper H. As is described below, both Gatekeeper H and endpoint B separately compute this keying material from the shared secret K_{BH} using a PRF.

Gatekeeper H shall generate a random Challenge-B, encryption key material EK_{BH} and salting key material KS_{BH} from the shared secret K_{BH} using the PRF-based key derivation procedure as defined in clause 12 where Challenge-B is substituted as **challenge** and CT_{HG}→**h235Key**→**secureSharedSecret**→**keyDerivationOID** shall hold "AnnexI-HMAC-SHA1-PRF", see clause 14.

EK_{GH} denotes the encryption key and KS_{GH} denotes the salting key that are shared between gatekeeper G and gatekeeper H. Gatekeeper H shall generate one random Challenge-G. Gatekeeper H shall generate encryption key material EK_{GH} and salting key material KS_{GH} from the shared secret K_{GH} using the PRF-based key derivation procedure as defined in clause 14 where Challenge-G is substituted for **challenge**. CT_{HG}→**challenge** shall hold challenge-G, the endpoint ID of the endpoint B shall be set in CT_{HG}→**h235Key**→**secureSharedSecret**→**generalID.**

Gatekeeper H shall transmit the encrypted EK_{BH} and the encrypted KS_{BH} to gatekeeper G. The enhanced OFB (EOFB) encryption mode (see 8.4/H.235.6) shall be used with the secret, endpoint-specific salting key KS_{GH}. Applicable encryption algorithms are (see Table 6/H.235.6):

– DES (56 bit) in EOFB mode using OID "Y1": optional;
– 3DES (168 bit) in outer-EOFB mode using OID "Z1": optional;
– AES (128 bit) in EOFB mode using OID "Z2": default and recommended;
– RC2-compatible (56 bit) in EOFB mode using OID "X1": optional.

For the EOFB encryption mode, gatekeeper H shall generate a random initial value IV. For OID "X1", OID "Y1" and OID "Z1" the IV has 64 bits and shall be conveyed within CT_{HG}→**h235Key**→**secureSharedSecret**→**params**→**iv8**; whereas the IV has 128 bits for OID "Z2" and shall be conveyed within CT_{HG}→**h235Key**→**secureSharedSecret**→**params**→**iv16**.

Gatekeeper H shall include $ENC_{EK_{GH}, KS_{GH}, IV}(EK_{BH})$ and $ENC_{EK_{GH}, KS_{GH}, IV}(KS_{BH})$ in ClearToken CT_{HG} with **tokenOID** set to "I13". The obtained ciphertext $ENC_{EK_{GH}, KS_{GH}, IV}(EK_{BH})$ shall be conveyed in CT_{HG}→**h235Key**→**secureSharedSecret**→**encryptedSessionKey**; the obtained ciphertext $ENC_{EK_{GH}, KS_{GH}, IV}(KS_{BH})$ shall be conveyed in CT_{HG}→**h235Key**→**secureSharedSecret**→**encryptedSaltingKey**. The encryption algorithm shall be indicated in CT_{HG}→**h235Key**→**algorithmOID** ("X1", "Y1", "Z1" or "Z2"). Challenge-B shall be placed within CT_{HG}→**h235Key**→**secureSharedSecret**→**clearSaltingKey**. CT_{HG}→**generalID** shall be set to the gatekeeper identifier G whereas CT_{HG}→**sendersID** shall be set to the gatekeeper identifier H.

Challenge-B shall be conveyed to endpoint B by inclusion of a **profileInfo** within the **ClearToken** CT_{HG}→**profileInfo→elementID** = 0 that identifies this particular profile element;

CT_{HG}→**profileInfo→paramS** left unused and CT_{HG}→**profileInfo→element→octets** shall hold Challenge-B.

The **LCF** response shall hold the ClearToken CT_{HG}.

9.5 ACF phase

The gatekeeper G, recognizing that endpoints A and B support this Recommendation, shall generate key material and ClearTokens as specified below.

The gatekeeper is able to calculate a call-based shared secret K_{AB}, besides the normal **ARQ** operation. This call-based shared secret is then propagated to both endpoints using ClearTokens. Those ClearTokens are conveyed within the **ACF** message and are sent back to the caller.

Two ClearTokens shall be included, one CT_A for the caller A and another one CT_B for the callee B. Each **ClearToken** shall contain an OID ("I11" or "I12") within **tokenOID** that indicates whether the token is destined for the caller (OID "I11" for CT_A) or for the callee (OID "I12" for CT_B).

GK G shall decrypt CT_{HG}→**h235Key→secureSharedSecret→encryptedSessionKey** to obtain EK_{BH} and shall decrypt CT_{HG}→**h235Key→secureSharedSecret→encryptedSaltingKey** to obtain KS_{BH}.

The **ClearToken** as defined in this Recommendation may be used in conjunction with other security profiles such as with H.235.1 or with H.235.3 that deploy ClearTokens as well. In such a case, ClearToken from this Recommendation shall use those other **ClearToken** fields too. For example, in order to use this Recommendation in conjunction with ITU-T Rec. H.235.1, the fields **timestamp**, **random**, **generalID**, **sendersID**, and **dhkey** shall be present and shall be used, as described by the H.235.1 security profiles.

The gatekeeper ID (GKID) of gatekeeper G shall be placed within CT_A→**sendersID** and within CT_B→**sendersID** whereas CT_A→**generalID** shall hold the endpoint identifier of endpoint A and CT_B→**generalID** the endpoint identifier of endpoint B.

Gatekeeper G shall generate salting key material KS_{GH} and encryption key material EK_{GH} from K_{GH} using the PRF-based key derivation procedure as defined in clause 12 with **challenge** substituted by CT_{HG}→**challenge.**

The encryption keys EK_{AG} and EK_{BH} for the encrypted end-to-end key K_{AB} shall be derived from the shared secret between the gatekeeper and the endpoints (EK_{AG} or EK_{BH}) using the PRF-based key derivation procedure as defined in clause 12 where both CT_A→**h235Key→secureSharedSecret→keyDerivationOID** and CT_B→**h235Key→secureSharedSecret→keyDerivationOID** shall hold "AnnexI-HMAC-SHA1-PRF", see clause 14 and CT_A→**challenge** shall hold Challenge-A.

Gatekeeper G shall copy Challenge-B from CT_{HG}→**h235Key→secureSharedSecret→clearSaltingKey** into CT_B→**challenge**.

CT_B→**profileInfo** shall hold the profile element that was conveyed in CT_{HG} **profileInfo** such that in the end endpoint B obtains Challenge-B.

This session secret K_{AB} shall be encrypted by EK_{AG} (for CT destined to endpoint A) or by EK_{BH} (for the CT destined to endpoint B) using an encryption algorithm.

The enhanced OFB (EOFB) encryption mode (see 8.4/H.235.6) shall be used with the secret, endpoint-specific salting key KS_{AG} or KS_{BH}. Applicable encryption algorithms are (see Table 6/H.235.6):

– DES (56 bit) in EOFB mode using OID "Y1": optional;

– 3DES (168 bit) in outer-EOFB mode using OID "Z1": optional;

– AES (128 bit) in EOFB mode using OID "Z2": default and recommended;

– RC2-compatible (56 bit) in EOFB mode using OID "X1": optional.

For the EOFB encryption mode, the gatekeeper G shall generate a random initial value IV. For OID "X1", OID "Y1" and OID "Z1" the IV has 64 bits and shall be conveyed within CT_A→**h235Key**→**secureSharedSecret**→**params**→**iv8** and within CT_B→**h235Key**→**secureSharedSecret**→**params**→**iv8**; whereas the IV has 128 bits for OID "Z2" and shall be conveyed within CT_A→**h235Key**→**secureSharedSecret**→**params**→**iv16** and within CT_B→**h235Key**→**secureSharedSecret**→**params**→**iv16**.

The obtained ciphertext $ENC_{EK_{AG}, KS_{AG}, IV}(K_{AB})$ shall be conveyed in CT_A→**h235Key**→**secureSharedSecret**→**encryptedSessionKey** and $ENC_{EK_{BH}, KS_{BH}, IV}(K_{AB})$ shall then be conveyed in CT_B→**h235Key**→**secureSharedSecret**→**encryptedSessionKey**. The encryption algorithm shall be indicated in CT_A→**h235Key**→**secureSharedSecret**→**algorithmOID** and in CT_B→**h235Key**→**secureSharedSecret**→**algorithmOID** ("X1", "Y1", "Z1" or "Z2").

For the ClearToken destined to endpoint A, the endpoint identifier of endpoint B ($EPID_B$) shall be placed within CT_A→**h235Key**→**secureSharedSecret**→**generalID**. Likewise for the ClearToken destined to endpoint B, the endpoint identifier of endpoint A ($EPID_A$) shall be placed within CT_B→**h235Key**→**secureSharedSecret**→**generalID**.

For the EOFB encryption algorithms, **encryptedSaltingKey** shall not be used.

The gatekeeper G shall include both ClearTokens CT_A and CT_B in the **ACF** towards endpoint A.

9.6 SETUP phase

Endpoint A shall identify CT_A by inspection of the **tokenOID** "I11" within ClearToken.

Endpoint A shall verify that the obtained CT_A is fresh by checking the **timestamp**. Further security checks shall verify the **generalID** and **sendersID** of the ClearToken and **generalID** within **V3KeySyncMaterial**. If the received CT_A was verified as being fresh, endpoint A shall retrieve the IV and compute EK_{AG} and KS_{AG} as described above for the gatekeeper G. Endpoint A shall decrypt the **encryptedSessionKey** information found within **secureSharedSecret** of CT_A to obtain K_{AB}.

If the received CT_A was verified as being fresh, endpoint A is able to send a SETUP message to endpoint B. This SETUP message includes CT_B. The SETUP message shall be secured (authenticated and/or integrity protected) according to ITU-T Rec. H.235.1 or according to ITU-T Rec. H.235.3 using K_{AB} as the applied shared secret. For this, **generalID** in the H.235.1 hashed ClearToken (not CT_B!) shall not be used unless endpoint A has already an $EPID_B$ available (e.g., through configuration or memorized from former communication). If endpoint A uses an $EPID_B$ value for generalID in SETUP, then endpoint A shall accept the value of the sendersID in the returned call signalling message as the true $EPID_B$.

Endpoint B shall identify CT_B by inspection of the **tokenOID** "I12" within ClearToken.

Endpoint B shall verify that the obtained CT_B is fresh by checking the **timestamp**. Further security checks shall verify the **sendersID** of the ClearToken and **generalID** within **secureSharedSecret**. If the received CT_B was verified as being fresh, endpoint B shall retrieve Challenge-B from CT_{HG}→**profileInfo**→**element**→**octets**, and retrieve the IV and compute EK_{BH} and KS_{BH}, Challenge-B substituted as **challenge** in clause 12 as described above for the gatekeeper. Endpoint B shall decrypt the **encryptedSessionKey** information found within **secureSharedSecret** of CT_B to obtain K_{AB}.

In the case where the CT_B is verified as being fresh, endpoint B is able to proceed with the call signalling by replying with CALL-PROCEEDING, ALERTING or CONNECT etc., as appropriate. In the case where the CT_B is found not to be fresh, or the security verification of the SETUP message failed, endpoint B shall reply with RELEASE-COMPLETE and the **ReleaseCompleteReason** set to a security error as defined by 11.1/H.235.0.

When media security is to be deployed (see 6.1/H.235.6), endpoint A and endpoint B shall exchange Diffie-Hellman half-keys according to 8.5/H.235.6 and establish a dynamic session-based master key from which media-specific session keys can then be derived.

Endpoint B shall include **generalID** set to $EPID_A$ and **sendersID** set to $EPID_B$ for protection of any H.225.0 Call signalling message destined to EP A (e.g., Call Proceeding, Alerting or Connect).

Figure 2 shows the basic communication flow:

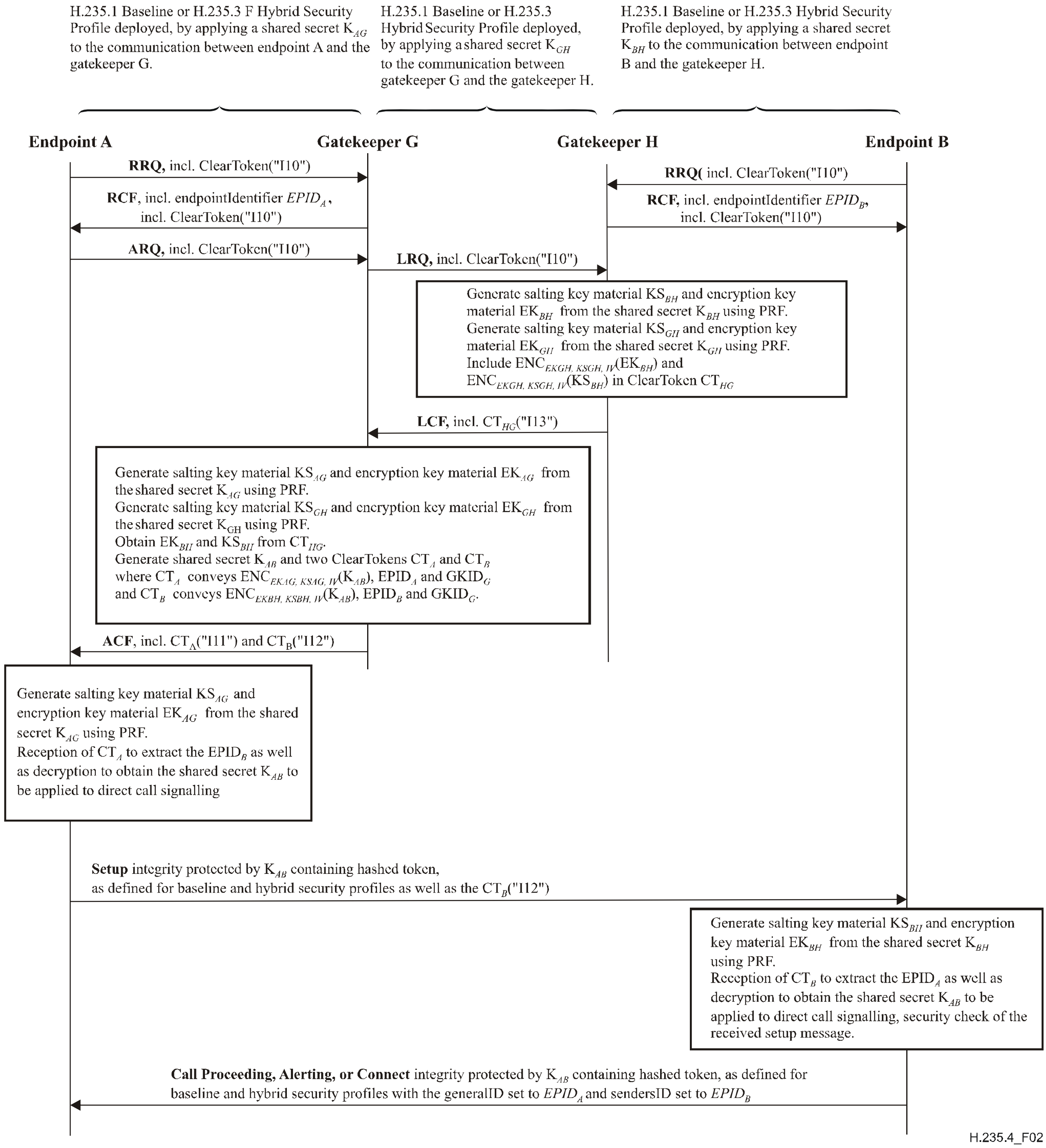

Figure 2/H.235.4 – Basic communication flow (DRC1)

10 Procedure DRC2 (interdomain environment)

The procedure described in this clause is applicable in interdomain environments where the gatekeepers are situated within different administrative domains and where each domain may employ a different security policy. Procedure DRC2 is applicable in cases where the calling endpoint or the gatekeepers do not support the Diffie-Hellman algorithm.

In such an environment, it is assumed acceptable that the terminating gatekeeper H determines the effective security policy for a call to be established; thus the terminating gatekeeper H selects and chooses the applied security parameters. The originating gatekeeper G will accept the chosen security parameters.

10.1 GRQ/RRQ phase

Endpoints capable of supporting this security profile shall indicate this fact during **GRQ** and/or **RRQ** by including a separate ClearToken with **tokenOID** set to "I20"; any other fields in that ClearToken should not be used. The H.235.4-capable gatekeeper that is willing to provide this functionality shall reply with **GCF** or **RCF** with a separate ClearToken included with **tokenOID** set to "I20" and all other fields in the ClearToken unused.

10.2 ARQ phase

Before an endpoint A starts sending call signalling messages to another endpoint B directly, the endpoint A or B shall apply for admission at the gatekeeper G or H using **ARQ**. Endpoint A shall include within **ARQ** a separate ClearToken with **tokenOID** set to "I20" and all other fields in the ClearToken unused.

10.3 LRQ phase

This procedure covers the case of both a single, common gatekeeper to the endpoints and the case of multiple, chained gatekeepers. In the case of multiple involved gatekeepers, gatekeeper G, in which zone the call originates, should locate gatekeeper H using the (multicast) **LRQ** mechanism as described by ITU-T Rec. H.323 clause 8.1.6, "Optional called endpoint signalling". The communication between two gatekeepers shall be secured according to ITU-T Rec. H.235.1. For this, it is assumed that a common shared secret K_{GH} is available. Since **LRQ** among gatekeepers is typically a multicast message, the shared secret K_{GH} typically cannot be a pair-wise shared secret but is assumed to be actually a group-based shared secret within the potential cloud of gatekeepers.

NOTE – This assumption limits scalability in the general case, and does not allow source authentication. However, it is believed that in corporate networks with a limited, small number of well-known gatekeepers, such constraint and security limitations are still acceptable. Securing inter-gatekeeper multicast communication using digital signatures could overcome those limitations: however, this remains for further study.

If the **LRQ** mechanism is used to locate the far-end gatekeeper, then **LRQ** shall convey a separate ClearToken with **tokenOID** set to "I20"; any other fields in that ClearToken should not be used. For the multicast case, the **generalID** in the ClearToken of **LRQ** shall not be used. Inter-gatekeeper communication using H.501 and/or H.510 remains for further study.

10.4 LCF phase

The gatekeeper H, recognizing that endpoints A and B support this Recommendation, shall generate key material and ClearTokens in **LCF** as specified below.

K_{BH} denotes the shared secret that is shared between endpoint B and gatekeeper H. EK_{BH} denotes the encryption key and KS_{BH} denotes the salting key that are shared between endpoint B and gatekeeper H. Gatekeeper H generates one random Challenge-B. Gatekeeper H shall generate encryption key material EK_{BH} from the shared secret K_{BH} using the PRF-based key derivation procedure with Challenge-B substituted as **challenge** as defined in clause 12 where CT_B→**h235Key→secureSharedSecret→keyDerivationOID** shall hold "AnnexI-HMAC-SHA1-PRF", see clause 14.

Gatekeeper H shall generate a salting key KS_{BH} from K_{BH} using the PRF-based key derivation procedure as defined in clause 12 with Challenge-B substituted as **challenge**.

EK_{GH} denotes the encryption key and KS_{GH} denotes the salting key that are shared between gatekeeper G and gatekeeper H. Gatekeeper H generates one random Challenge-G. Gatekeeper H shall generate encryption key material EK_{GH} from the shared secret K_{GH} using the PRF-based key derivation procedure with Challenge-G substituted as **challenge** as defined in clause 12 where CT_{HG}→**h235Key→secureSharedSecret→keyDerivationOID** shall hold "AnnexI-HMAC-SHA1-PRF", see clause 14.

Gatekeeper H shall generate KS_{GH} from the shared secret K_{GH} using the PRF-based key derivation procedure with Challenge-G substituted as **challenge** as defined in clause 12.

Gatekeeper H creates two ClearTokens in the **LCF** message. One CT_{HG} for the Gatekeeper G and a CT_B for the callee B. CT_{HG}→**tokenOID** shall contain an OID "I23" whereas CT_B→**tokenOID** shall contain OID "I12".

Challenge-G shall be set in CT_{HG}→**challenge**, the gatekeeper ID of the Gatekeeper H shall be set in CT_{HG}→**sendersID**, the gatekeeper ID of the Gatekeeper G (copied from the **LRQ**) shall be set in CT_{HG}→**generalID**.

Challenge-B shall be set in CT_B→**challenge**, the gatekeeper ID of the Gatekeeper H shall be set in CT_B→**sendersID**, the endpoint ID of the endpoint B shall be set in CT_B→**generalID**. If the **LRQ** has the endpoint ID of endpoint A in **LRQ's** endpointIdentifier field, Gatekeeper H shall copy it into CT_B→**h235Key→secureSharedSecret→generalID**, and shall also copy it into CT_{HG}→**h235Key→secureSharedSecret→generalID** too.

The **LCF** response shall hold the ClearToken CT_{HG} and CT_B if Gatekeeper H and endpoint B support DRC2 of this Recommendation too.

Gatekeeper G having received the **LCF** message from Gatekeeper H, checks the ClearToken CT_B and CT_{HG}. Gatekeeper G uses Challenge-G as **challenge** and the PRF as in clause 12 to compute KS_{GH} and EK_{GH} from K_{GH} and then to decrypt CT_{HG}→**h235Key→secureSharedSecret→encryptedSessionKey** and obtains the K_{AB} shared by endpoints A and B.

10.5 ACF phase

The gatekeeper H calculates a call-based shared secret K_{AB} that is shared by endpoints A and B. This call-based shared secret is then propagated to both endpoints using ClearTokens. The ClearToken is first sent back to the originating gatekeeper G and then Gatekeeper G conveys the information within the **ACF** message back to the caller.

Gatekeeper H shall encrypt the K_{AB} by EK_{GH} as $ENC_{EK_{HG}, KS_{HG}, IV}(K_{AB})$ and put the encrypted K_{AB} into CT_{HG}→**h235Key→secureSharedSecret→encryptedSessionKey**.

The enhanced OFB (EOFB) encryption mode (see 8.4/H.235.6) shall be used with the secret, endpoint-specific salting key KS_{GH}. Applicable encryption algorithms are (see Table 6/H.235.6):

– DES (56 bit) in EOFB mode using OID "Y1": optional;

– 3DES (168 bit) in outer-EOFB mode using OID "Z1": optional;

– AES (128 bit) in EOFB mode using OID "Z2": default and recommended;

– RC2-compatible (56 bit) in EOFB mode using OID "X1": optional.

For the EOFB encryption mode, Gatekeeper H shall generate a random initial value IV. For OID "X1", OID "Y1" and OID "Z1" the IV has 64 bits and shall be conveyed within CT_{HG}→**h235Key**→**secureSharedSecret**→ **params**→**iv8**; whereas the IV has 128 bits for OID "Z2" and shall be conveyed within CT_{HG}→**h235Key**→**secureSharedSecret**→**params**→**iv16**.

The encryption algorithm shall be indicated in CT_{HG}→**h235Key**→**secureSharedSecret**→**algorithmOID** ("X1", "Y1", "Z1" or "Z2"). For the EOFB encryption algorithms, **encryptedSaltingKey** shall not be used.

Likewise, Gatekeeper H shall encrypt the K_{AB} by EK_{BH} as $ENC_{EK_{BH}, KS_{BH}, IV}(K_{AB})$ and put that encrypted K_{AB} into CT_B→**h235Key**→**secureSharedSecret**→**encryptedSessionKey**.

The enhanced OFB (EOFB) encryption mode (see 8.4/H.235.6) shall be used with the secret, endpoint-specific salting key KS_{BH} for endpoint B (CT_B). Applicable encryption algorithms are (see Table 6/H.235.6):

– DES (56 bit) in EOFB mode using OID "Y1": optional;

– 3DES (168 bit) in outer-EOFB mode using OID "Z1": optional;

– AES (128 bit) in EOFB mode using OID "Z2": default and recommended;

– RC2-compatible (56 bit) in EOFB mode using OID "X1": optional.

For the EOFB encryption mode, Gatekeeper H shall generate a random initial value IV. For OID "X1", OID "Y1" and OID "Z1" the IV has 64 bits and shall be conveyed within CT_B→**h235Key**→**secureSharedSecret**→ **params**→**iv8**; whereas the IV has 128 bits for OID "Z2" and shall be conveyed within CT_B→**h235Key**→**secureSharedSecret**→**params**→**iv16**.

The encryption algorithm shall be indicated in CT_B→**h235Key**→**secureSharedSecret**→**algorithmOID** ("X1", "Y1", "Z1" or "Z2"). For the EOFB encryption algorithms, **encryptedSaltingKey** shall not be used.

For the **ACF** response to endpoint A, two ClearTokens shall be included, one CT_A for the caller A and another one CT_B for the callee B. **ClearToken** CT_A→**tokenOID** shall contain an OID "I11".

Gatekeeper G generates one Challenge-A, and generates encryption key material EK_{AG} from the shared secret K_{AG} using the PRF-based key derivation procedure with Challenge-A substituted as **challenge** using the PRF-based key derivation procedure as defined in clause 12 where CT_A→**h235Key**→**secureSharedSecret**→**keyDerivationOID** shall hold "AnnexI-HMAC-SHA1-PRF", see clause 14 and sets CT_A→**challenge** to Challenge-A.

Gatekeeper G shall encrypt K_{AB} by EK_{AG} as $ENC_{EK_{AG}, KS_{AG}, IV}(K_{AB})$ using an encryption algorithm and put the encrypted K_{AB} into CT_A→**h235Key**→**secureSharedSecret**→**encryptedSessionKey**.

The enhanced OFB (EOFB) encryption mode (see 8.4/H.235.6) shall be used with the secret, endpoint-specific salting key KS_{AG}. Applicable encryption algorithms are (see Table 6/H.235.6):

– DES (56 bit) in EOFB mode using OID "Y1": optional;

– 3DES (168 bit) in outer-EOFB mode using OID "Z1": optional;

– AES (128 bit) in EOFB mode using OID "Z2": default and recommended;

– RC2-compatible (56 bit) in EOFB mode using OID "X1": optional.

For the EOFB encryption mode, the GK G shall generate a random initial value IV. For OID "X1", OID "Y1" and OID "Z1" the IV has 64 bits and shall be conveyed within CT_A→**h235Key**→**secureSharedSecret**→**params**→**iv8**; whereas the IV has 128 bits for OID "Z2" and shall be conveyed within CT_A→**h235Key**→**secureSharedSecret**→**params**→**iv16**. The encryption algorithm shall be indicated in CT_A→**h235Key**→**secureSharedSecret**→**algorithmOID** ("X1", "Y1", "Z1" or "Z2").

The gatekeeper ID of the Gatekeeper G shall be set in CT_A→**sendersID**, the endpoint ID of the endpoint A shall be set in CT_A→**generalID**. The endpoint ID of endpoint B shall be copied from CT_B→**generalID** into CT_A→**h235Key**→**secureSharedSecret**→**generalID.**

If Gatekeeper G has not filled the endpoint ID of endpoint A in **LRQ**'s endpointIdentifier field before, Gatekeeper G shall fill the endpoint ID of endpoint A into CT_B→**h235Key**→**secureSharedSecret**→**generalID**.

For the EOFB encryption algorithms, **encryptedSaltingKey** shall not be used.

The **ClearToken**, as defined in this Recommendation, may be used in conjunction with other security profiles such as H.235.1 or H.235.3 that deploy ClearTokens as well. In such a case, the ClearToken of this Recommendation shall use those other **ClearToken** fields too. For example, in order to use this Recommendation in conjunction with ITU-T Rec. H.235.1, the fields **timestamp**, **random**, **generalID**, **sendersID**, and **dhkey** shall be presented and shall be used, as described by the H.235.1 security profiles.

The gatekeeper ID (GKID) of gatekeeper G shall be placed within CT_A→**sendersID** whereas CT_A→**generalID** shall hold the endpoint identifier of endpoint A.

Endpoint A shall identify CT_A by inspection of the CT_A→**tokenOID** "I21". Endpoint A shall verify that the obtained CT_A is fresh by checking the **timestamp**. Further security checks shall verify the **generalID** and **sendersID** of the ClearToken and **generalID** within **secureSharedSecret**. If the received CT_A was verified as being fresh, endpoint A shall retrieve the IV and compute EK_{AG} and KS_{AG} as described above for the gatekeeper G using CT_A→**challenge** as Challenge-A substituted as **challenge** within clause 12. Endpoint A shall decrypt CT_A→**h235Key**→**secureSharedSecret**→**encryptedSessionKey** to obtain K_{AB}.

10.6 SETUP phase

Endpoint A shall identify CT_A by inspection of the CT_A →**tokenOID** "I11". Endpoint A shall verify that the obtained CT_A is fresh by checking the timestamp. Further security checks shall verify the **generalID** and **sendersID** of the ClearToken and **generalID** within **secureSharedSecret**. If the received CT_A was verified as being fresh, endpoint A shall retrieve the IV and compute EK_{AG} and KS_{AG} as described above for the gatekeeper G using CT_A→**challenge** as Challenge-A. Endpoint A shall decrypt CT_A→**h235Key**→**secureSharedSecret**→**encryptedSessionKey** to obtain K_{AG}.

If the received CT_A is verified as being fresh, endpoint A is able to send a SETUP message to endpoint B. This SETUP message includes CT_B. The SETUP message shall be secured (authenticated and/or integrity protected) according to ITU-T Rec. H.235.1 or ITU-T Rec. H.235.3 using K_{AB} as the applied shared secret. For this, **generalID** in the H.235.1 hashed ClearToken (not CT_B!) shall not be used unless endpoint A already has an $EPID_B$ available (e.g., through configuration or memorized from former communication). If endpoint A uses an $EPID_B$ value for generalID in SETUP, then endpoint A shall accept the value of the sendersID in the returned call signalling message as the true $EPID_B$.

Endpoint B shall identify CT_B by inspection of the **tokenOID** "I12" within ClearToken.

Endpoint B shall verify that the obtained CT_B is fresh by checking the **timestamp**. Further security checks shall verify the **sendersID** of the ClearToken and **generalID** within **secureSharedSecret**. If the received CT_B was verified as being fresh, endpoint B shall retrieve the IV, compute EK_{BH} and KS_{BH}, using CT_B→**challenge** as Challenge-B substituted as **challenge** in clause 12 as described above for the gatekeeper H. Endpoint B shall decrypt CT_B→**h235Key**→**secureSharedSecret**→**encryptedSessionKey** to obtain K_{AB}.

In the case where the CT_B is verified as being fresh, endpoint B is able to proceed the call signalling by replying with CALL-PROCEEDING, ALERTING or CONNECT etc., as appropriate. In the case where the CT_B is found not to be fresh, or the security verification of the SETUP message fails, endpoint B shall reply with RELEASE-COMPLETE and the **ReleaseCompleteReason** set to a security error as defined by 11.1/H.235.0.

When media security is to be deployed (see 6.1/H.235.6), endpoint A and endpoint B shall exchange Diffie-Hellman half-keys according to 8.5/H.235.6 and establish a dynamic session-based master key from which media-specific session keys can then be derived.

Endpoint B shall include **generalID** set to $EPID_A$ and **sendersID** set to $EPID_B$ for protection of any H.225.0 Call signalling message destined to EP A (e.g., Call Proceeding, Alerting or Connect).

Figure 3 shows the basic communication flow:

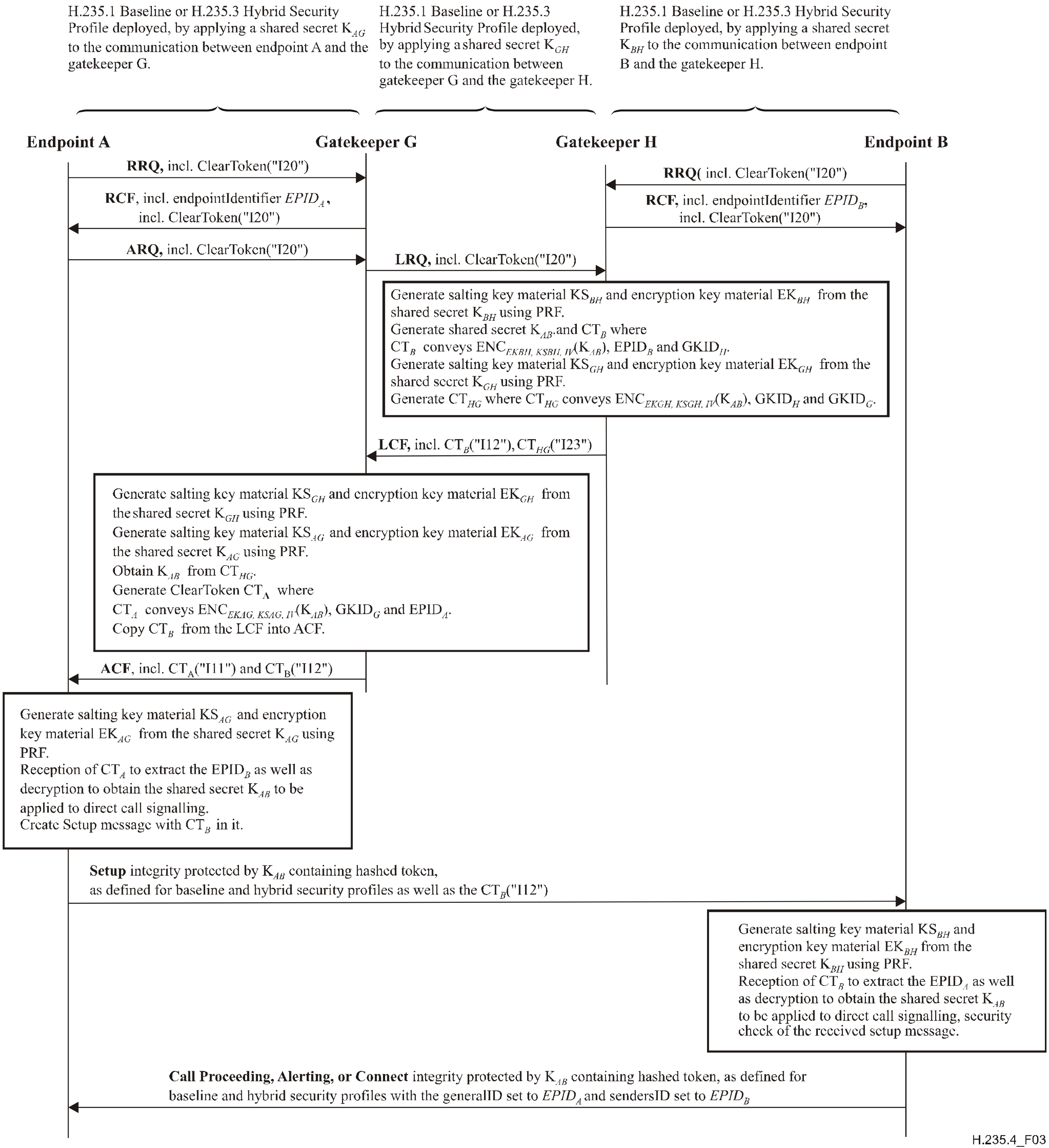

Figure 3/H.235.4 – Basic communication flow (DRC2)

11 Procedure DRC3 (interdomain environment)

The procedure described in this clause is applicable in interdomain environments and where the calling endpoint does not support the Diffie-Hellman algorithm while the Gatekeepers in the calling and called domain both are capable of computing DH exchange. In such an environment, the session key is computed by exchanging DH parameters between originating gatekeeper and terminating gatekeeper.

11.1 GRQ/RRQ phase

The scenario covers multiple, chained gatekeepers. Endpoints capable of supporting this security profile shall indicate this fact during **GRQ** and/or **RRQ** by including a separate ClearToken with **tokenOID** set to "I30"; any other fields in that ClearToken are unused. The H.235.4-capable gatekeeper that is willing to provide this functionality shall reply with **GCF** or **RCF** with a separate ClearToken included with **tokenOID** set to "I30" and all other fields in the ClearToken unused.

11.2 ARQ phase

Before EP A calls EP B using DRC3, EP A sends an **ARQ** message to GK G and the **ARQ** message contains a separate ClearToken with **tokenOID** set to "I30" and other fields unused.

11.3 LRQ phase

On the reception of the **ARQ** message sent by EP A, GK G sends **LRQ** to GK H to inquire EP B's address since EP B does not belong to GK G's domain. GK G checks the ClearToken carried by the **ARQ** message finding that **tokenOID** is set to "I30", if GK G supports the DH algorithm, then it applies some pre-configured rules which determine that DRC3 should be chosen.

Then GK G generates a **LRQ** message containing a ClearToken (within the CryptoHashedToken) with its **tokenOID** set to "I30" to indicate to GK H that a DH key negotiation is needed. The **dhkey** field of the ClearToken is filled with the caller's DH parameters (g, p, g^x) generated by GK G and other fields are unused.

GK G then sends this **LRQ** message to GK H. In the case of GK cloud, GK G sends the **LRQ** message to its immediately neighbouring GK which in turn forwards the **LRQ** message to its own immediately neighbouring GK. The forwarding process continues until the **LCF** message finally reaches GK H.

For the multicast case, the **generalID** in the CryptoToken of **LRQ** shall not be used. If GK G was not able to locate the far-end endpoint B then GK G shall return **ARJ** to endpoint A. The communication between two gatekeepers shall be secured according to ITU-T Rec. H.235.1.

If GK G does not support the profile, GK G is free to choose whether to fall back to DRC2, or return ARJ to endpoint A. If DRC2 is chosen, all subsequent phases including the **LRQ** phase are the same as those of DRC2.

11.4 LCF phase

After receiving the **LRQ** message from GK G, the GK H, recognizing that both endpoint A and B support this procedure, shall generate the session key K_{AB} as specified below.

Firstly, GK H produces a random Challenge-B, which shall be set to CT_B→**challenge** and CT_B→**h235Key→secureSharedSecret→keyDerivationOID** shall hold "AnnexI-HMAC-SHA1-PRF", and then uses the shared key K_{GH} and the Challenge-B to derive the key material EK_{GH} and the salting key KS_{GH} using PRF-based key derivation procedure.

Challenge-B shall be set in CT_B→**challenge**, the gatekeeper ID of the GK H shall be set in CT_B→**sendersID**, the endpoint ID of the EP B shall be set in CT_B→**generalID**. If **LRQ** has the endpoint ID of EP A in **LRQ's** endpoint Identifier field, Gatekeeper H shall copy it into

CT_B→**h235Key→secureSharedSecret→generalID**, and shall also copy it into CT_{HG}→**h235Key→secureSharedSecret→generalID** too.

GK H then creates two ClearTokens in the **LCF** message. One CT_{HG} for GK G whose **tokenOID** is set to "I33" and one CT_B for EP B whose **tokenOID** is set to "I12". GK H generates the callee's DH parameters (g, p, g^y). With the caller's DH parameters obtained from the **LRQ** message, GK H shall compute the session key $K_{AB} = g^{xy}$ mod p.

Finally, GK H shall encrypt K_{AB} using EK_{BH} and KS_{BH} as $ENC_{EK_{BH}, KS_{BH}, IV}(K_{AB})$ and put the encrypted K_{AB} into CT_B→**h235Key→secureSharedSecret→encryptedSessionKey**, and puts the callee's DH parameters into **dhkey** of CT_{HG}.

The enhanced OFB (EOFB) encryption mode (see 8.4/H.235.6) shall be used with the secret, endpoint-specific salting key KS_{GH}. Applicable encryption algorithms are (see Table 6/H.235.6):

– DES (56 bit) in EOFB mode using OID "Y1": optional;

– 3DES (168 bit) in outer-EOFB mode using OID "Z1": optional;

– AES (128 bit) in EOFB mode using OID "Z2": default and recommended;

– RC2-compatible (56 bit) in EOFB mode using OID "X1": optional.

For the EOFB encryption mode, Gatekeeper H shall generate a random initial value IV. For OID "X1", OID "Y1" and OID "Z1" the IV has 64 bits and shall be conveyed within CT_B→**h235Key→secureSharedSecret→ params→iv8**; whereas the IV has 128 bits for OID "Z2" and shall be conveyed within CT_B→**h235Key→secureSharedSecret→params→iv16**.

The encryption algorithm shall be indicated in CT_B→**h235Key→secureSharedSecret→algorithmOID** ("X1", "Y1", "Z1" or "Z2"). For the EOFB encryption algorithms, **encryptedSaltingKey** shall not be used.

GK H sends the **LCF** message to GK G. If GK cloud is present, the **LCF** message is transferred in a relay manner. Along this path, each GK receives the **LCF** message from its upstream immediate neighbour and checks the **LCF** message containing CT_{HG} and forwards the **LCF** message to its downstream immediate neighbour.

If GK H does not support the DH algorithm, or security policy is not allowed for DRC3, a fallback to DRC2 will occur. Therefore, the **LCF** phase and all the subsequent phases are the same as those of DRC2.

11.5 ACF phase

After receiving the **LCF** message, GK G, recognizing **tokenOID** in the separate ClearToken is set to "I33", obtains callee's DH and creates a ClearToken denoted CT_A with its **tokenOID** set to "I11" by means specified below.

Firstly, GK G produces a random Challenge-A, which shall be set to CT_A→**challenge** and CT_A→**h235Key→secureSharedSecret→keyDerivationOID** shall hold "AnnexI-HMAC-SHA1-PRF", and then uses the shared key K_{AG} and the Challenge-A to derive the key material EK_{AG} and the salting key KS_{AG} using PRF-based key derivation procedure.

Secondly, GK G uses caller's DH parameters which are retained in the **LRQ** phase and, in conjunction with callee's DH parameters, computes the session key $K_{AG} = g^{xy}$ mod p.

Then GK G copies the ClearToken CT_B from the **LCF** message to the **ACF** message whose **tokenOID** is set to "I12".

Finally, GK G encrypts K_{AB} using EK_{AG} and KS_{AG} as $ENC_{EK_{AG}, KS_{AG}, IV}(K_{AB})$ and puts the encrypted K_{AB} into CT_A→**h235Key→secureSharedSecret→encryptedSessionKey**, and copies CT_B from the **LCF** message into the **ACF** message.

The enhanced OFB (EOFB) encryption mode (see 8.4/H.235.6) shall be used with the secret, endpoint-specific salting key KS_{AG}.

Applicable encryption algorithms are (Table 6/H.235.6):

– DES (56 bit) in EOFB mode using OID "Y1": optional;
– 3DES (168 bit) in outer-EOFB mode using OID "Z1": optional;
– AES (128 bit) in EOFB mode using OID "Z2": default and recommended;
– RC2-compatible (56 bit) in EOFB mode using OID "X1": optional.

For the EOFB encryption mode, the GK G shall generate a random initial value IV. For OID "X1", OID "Y1" and OID "Z1" the IV has 64 bits and shall be conveyed within CT_A→**h235Key→secureSharedSecret→params→iv8**; whereas the IV has 128 bits for OID "Z2" and shall be conveyed within CT_A→**h235Key→secureSharedSecret→params→iv16**. The encryption algorithm shall be indicated in CT_A→**h235Key→secureSharedSecret→algorithmOID** ("X1", "Y1", "Z1" or "Z2").

If it is found that the ClearToken (within **LCF**) **tokenOID** is "I23", it can be judged that a fallback to DRC2 has occurred, and GK G is free to choose whether to accept GK H's security policy. If it accepts, the **ACF** phase and the subsequent Setup phase will be the same as those of DRC2. Otherwise, respond with a corresponding reject message indicating security failure by setting the reject reason to securityDenial.

GK G sends the **ACF** message to EP A.

11.6 SETUP phase

Endpoint A shall identify CT_A by inspection of the CT_A →**tokenOID** "I11". Endpoint A shall verify that the obtained CT_A is fresh by checking the timestamp. Further security checks shall verify the **generalID** and **sendersID** of the ClearToken and **generalID** within **secureSharedSecret**. If the received CT_A is verified as being fresh, endpoint A shall retrieve the IV and compute EK_{AG} and KS_{AG} as described above for the gatekeeper G using CT_A→**challenge** as Challenge-A. Endpoint A shall decrypt CT_A→**h235Key→secureSharedSecret→encryptedSessionKey** to obtain K_{AG}.

If the received CT_A is verified as being fresh, endpoint A is able to send a SETUP message to endpoint B. This SETUP message includes CT_B. The SETUP message shall be secured (authenticated and/or integrity protected) according to ITU-T Rec. H.235.1 or ITU-T Rec. H.235.3 using K_{AB} as the applied shared secret. For this, **generalID** in the H.235.1 hashed ClearToken (not CT_B!) shall not be used unless endpoint A already has an $EPID_B$ available (e.g., through configuration or memorized from former communication). If endpoint A uses an $EPID_B$ value for **generalID** in SETUP, then endpoint A shall accept the value of the **sendersID** in the returned call signalling message as the true $EPID_B$.

Endpoint B shall identify CT_B by inspection of the **tokenOID** "I12" within ClearToken.

Endpoint B shall verify that the obtained CT_B is fresh by checking the timestamp. Further security checks shall verify the **sendersID** of the ClearToken and **generalID** within **secureSharedSecret**. If the received CT_B is verified as being fresh, endpoint B shall retrieve the IV, compute EK_{BH} and KS_{BH}, using CT_B→**challenge** as Challenge-B. Endpoint B shall decrypt CT_B→**h235Key→secureSharedSecret→encryptedSessionKey** to obtain K_{AB}.

In the case where the CT_B is verified as being fresh, endpoint B is able to proceed the call signalling by replying with CALL-PROCEEDING, ALERTING or CONNECT etc., as appropriate. In the case where the CT_B is found not to be fresh, or the security verification of the SETUP message fails, endpoint B shall reply with RELEASE-COMPLETE and the **ReleaseCompleteReason** set to a security error as defined by 11.1/H.235.0.

When media security is to be deployed (see 6.1/H.235.6), endpoint A and endpoint B shall exchange Diffie-Hellman half-keys according to 8.5/H.235.6 and establish a dynamic session-based master key from which media-specific session keys can then be derived.

Endpoint B shall include **generalID** set to $EPID_A$ and **sendersID** set to $EPID_B$ for protection of any H.225.0 Call signalling message destined to EP A (e.g., Call Proceeding, Alerting or Connect).

Figure 4 shows the basic communication flow:

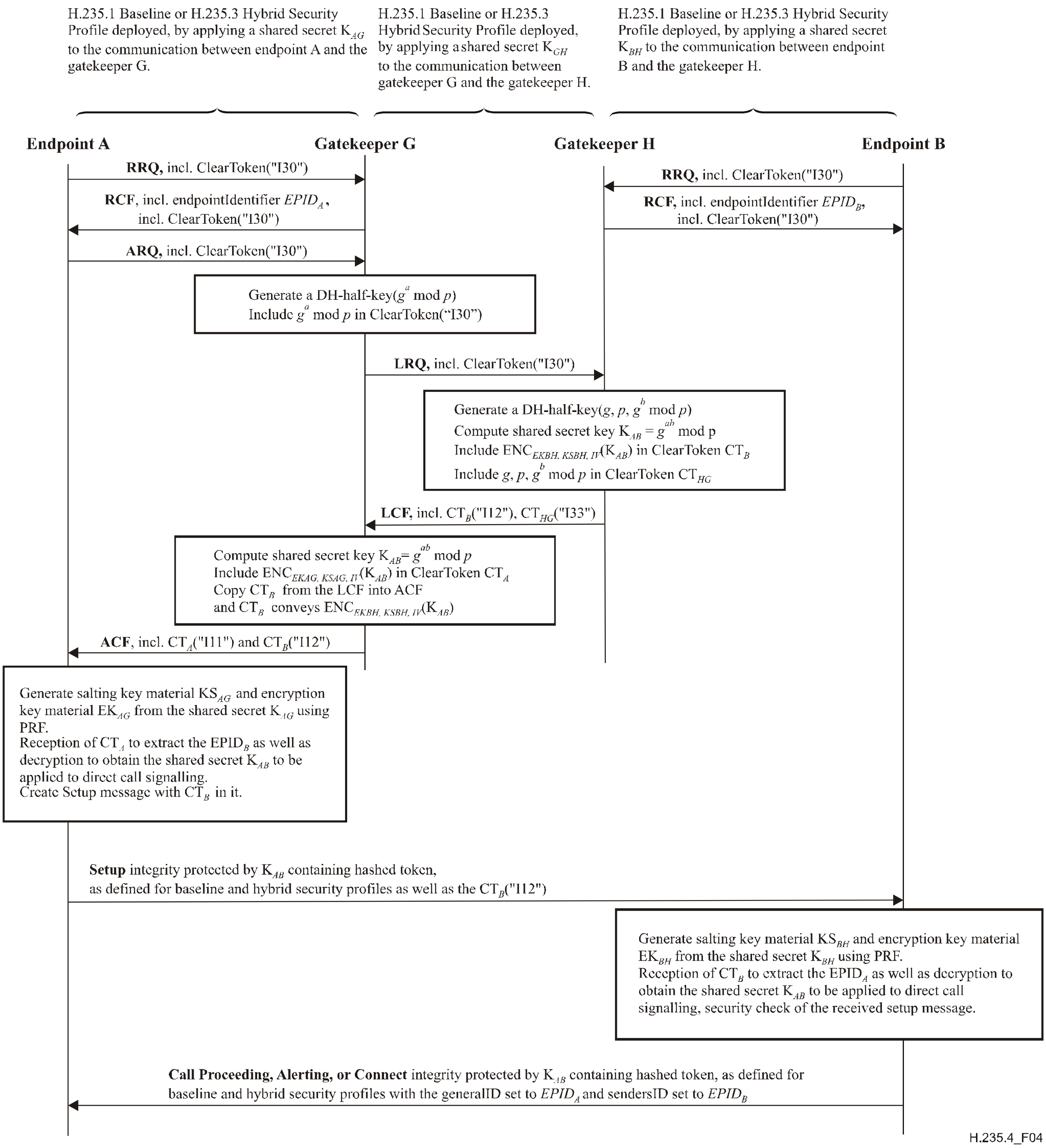

Figure 4/H.235.4 – Communication flow in DRC3

12 PRF-based key derivation procedure

This clause describes a procedure that defines how to derive key material from the shared secret and other parameters.

The procedure in this clause allows computing an encryption key and a salting key from a shared key. The procedure is uniform irrespective of the shared secret (K_{AG}, K_{BH} or K_{GH}).

In order to obtain the target keying material (e.g., EK_{AG}), the PRF (see clause 10/H.235.0) shall be used with the parameters taken from Table 1 where the *inkey* parameter is set to the corresponding shared key (e.g., K_{AG}), and *label* shall be set to the corresponding constant (e.g., 0x2AD01C64 || **challenge-A)** where || denotes concatenation. The *outkey_len* shall be set to the length of the required length of the target key material which depends on the chosen encryption algorithm.

NOTE – For EK_{AG}, KS_{AG}, EK_{BH} and KS_{BH} the 32-bit constant integers (i.e., 0x2AD01C64 etc.) are taken from the decimal digits of *e* (i.e., 2.71828...), and for EK_{GH} and KS_{GH}, the 32-bit constants integers are taken from the decimal digits of π (i.e., 3.14159...). For EK_{AG}, EK_{BH}, KS_{AG}, and KS_{BH}, the 32-bit integers are from blocks of 9 decimal digits, respectively the first, second, fourth and seventh blocks. The value for EK_{GH} comes from the first 10 decimal digits of π, while KS_{GH} comes from the subsequent 8 decimal digits of π.

Table 1/H.235.4 – Calculating encryption and salting keys from a shared secret

Target Key	PRF inkey	Constant || challenge
EK_{AG}	K_{AG}	0x2AD01C64 || **Challenge-A**
KS_{AG}	K_{AG}	0x150533E1 || **Challenge-A**
EK_{BH}	K_{BH}	0x1B5C7973 || **Challenge-B**
KS_{BH}	K_{BH}	0x39A2C14B || **Challenge-B**
EK_{GH}	K_{GH}	0x54655307 || **Challenge-G**
KS_{GH}	K_{GH}	0x35855C60 || **Challenge-G**

13 FIPS-140-based key derivation procedure

This clause may describe a procedure that defines how to derive key material from a shared secret and other parameters using a FIPS-140 compliant crypto module. This remains for further study.

14 List of object identifiers

Table 2/H.235.4 – Object identifiers used by H.235.4

Object identifier reference	Object identifier value	Description
"I10"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 48}	Used in procedure DRC1 during GRQ/RRQ and GCF/RCF and ARQ to let the EP/GK indicate support of DRC1.
"I11"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 49}	Used in procedures DRC1, DRC2 and DRC3 for the ClearToken tokenOID indicating that the ClearToken CT_A holds an end-to-end key for the caller.
"I12"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 50}	Used in procedures DRC1, DRC2 and DRC3 for the ClearToken tokenOID indicating that the ClearToken CT_B holds an end-to-end key for the callee.
"I13"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 52}	Used in procedure DRC1 for the inter-gatekeeper ClearToken tokenOID indicating that the ClearToken CT_{HG} holds an encryption key for the originating gatekeeper.
"I20"	{itu-t (0) recommendation (0) h (8) 235 version (0) 4 53}	Used in procedure DRC2 during GRQ/RRQ and GCF/RCF and ARQ to let the EP/GK indicate support of DRC2.
"I23"	{itu-t (0) recommendation (0) h (8) 235 version (0) 4 56}	Used in procedure DRC2 for the inter-gatekeeper ClearToken CT_{HG} tokenOID indicating that the ClearToken holds an encryption key for the originating gatekeeper.
"I30"	{itu-t (0) recommendation (0) h (8) 235 version (0) 4 34}	For use in separate ClearToken in GRQ/RRQ, GCF/RCF, ARQ to indicate support for DRC3. For use in separate ClearToken in LRQ to indicate carrying caller's DH parameters.
"I33"	{itu-t (0) recommendation (0) h (8) 235 version (0) 4 37}	For use in separate ClearToken in LCF to indicate carrying callee's DH parameters.
"Annex I -HMAC-SHA1-PRF"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 51}	Used in procedures DRC1, DRC2 and DRC3 for keyDerivationOID within V3KeySyncMaterial to indicate the applied key derivation method in clause 12 using the HMAC-SHA1 pseudo-random function.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	**Audiovisual and multimedia systems**
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects and next-generation networks
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2006