International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

H.235

Annex G

(01/2005)

SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS

Infrastructure of audiovisual services – Systems aspects

Security and encryption for H-series (H.323 and other H.245-based) multimedia terminals

Annex G: Usage of the MIKEY key management protocol for the secure real time transport protocol (SRTP) within H.235

ITU-T Recommendation H.235 – Annex G

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

Security and encryption for H-series (H.323 and other H.245-based) multimedia terminals

Annex G

Usage of the MIKEY key management protocol for the secure real time transport protocol (SRTP) within H.235

Summary

The purpose of Annex G to ITU-T Rec. H.235 is to provide recommendations of security procedures for H.323/H.235-based systems to use the IETF MIKEY key management protocol in conjunction with the IETF SRTP security protocol.

This annex is written as a security profile of H.235 that is offered as an option and may complement the other media security features of H.235 (Annex B, Annex D.7).

H.235 Annex G enables deploying SRTP media security where the MIKEY key management supplies the necessary keys and security parameters among the involved endpoints end-to-end. Annex G can be deployed within a H.323 domain among H.235 Annex-G-enabled H.323 systems. The annex defines the security protocol extensions to H.225.0 RAS and Call Signalling as well as H.245 along with the corresponding procedures. Furthermore, this annex provides the capabilities to support interworking with IETF SIP entities that have implemented the MIKEY key management and SRTP.

Source

Annex G to ITU-T Recommendation H.235 was approved on 8 January 2005 by ITU-T Study Group 16 (2005-2008) under the ITU-T Recommendation A.8 procedure.

Keywords

Media encryption, MIKEY key management, multimedia security, security profile, secure Real Time Transport Protocol, SRTP.

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

Security and encryption for H-series (H.323 and other H.245-based) multimedia terminals

Annex G

Usage of the MIKEY key management protocol for the secure real time transport protocol (SRTP) within H.235

G.1 Scope

The purpose of this annex to ITU-T Rec. H.235 is to provide recommendations of security procedures for H.323/H.235-based systems to use the MIKEY key management protocol in conjunction with the SRTP security protocol.

This security profile is offered as an option and may complement the other media security features of H.235 (Annex B, Annex D.7).

G.2 References

G.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.

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

[H.235] ITU-T Recommendation H.235 version 3 (2003), *Security and encryption for H-series (H.323 and other H.245-based) multimedia terminals,* plus Amd.1 (2004).

[H.245] ITU-T Recommendation H.245 (2005), *Control protocol for multimedia communication*.

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

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

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

[RFC 3550] H. Schulzrinne, S. Casner *et al*: RTP: A Transport Protocol for Real-Time Applications, *RFC 3550, IETF,* 07/2003.

[RFC 3711] M. Baugher *et al*: The Secure Real Time Transport Protocol, *RFC 3711, IETF,* 03/2004.

[RFC 3830] J. Arkko, E. Carrara, F. Lindholm, M. Naslund, K. Norrman: MIKEY: Multimedia Internet KEYing, *RFC 3830, IETF,* 08/2004.

G.2.2 Non-normative references and bibliography

[RFC 1305] D. Mills: Network Time Protocol (Version 3) Specification, Implementation and Analysis, *RFC 1305, IETF,* March 1992.

[RFC 2327] M. Handley, V. Jacobson: SDP: Session Description Protocol, *RFC 2327, IETF,* April 1998.

[RFC 2631] E. Rescorla: Diffie-Hellman Key Agreement Method, *RFC 2631, IETF,* June 1999.

[RFC 3261] J. Rosenberg *et al*: SIP: Session Initiation Protocol, *RFC 3261, IETF,* June 2002.

[RFC 3264] J. Rosenberg and H. Schulzrinne: An Offer/Answer Model with Session Description Protocol (SDP), *RFC 3264, IETF,* June 2002.

[SDP-New] M. Handley, Van Jacobson, C. Perkins: SDP: Session Description Protocol, *draft-ietf-mmusic-sdp-new-24.txt, IETF,* 02/2005.

[KMGMT-ext] J. Arkko, E. Carrara *et al*: Key Management Extensions for Session Description Protocol (SDP) and Real Time Streaming Protocol (RTSP), *Internet Draft draft-ietf-mmusic-kmgmt-ext-14.txt, Work in Progress, IETF,* 03/2005.

[MIKEY-DHHMAC] M. Euchner: HMAC-authenticated Diffie-Hellman for MIKEY, *Internet Draft draft-ietf-msec-MIKEY-DHHMAC-11.txt, Work in Progress, IETF,* 04/2005.

G.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, and X.800 apply along with those in this clause.

G.3.1 crypto session bundle (CSB): Collection of one or more Crypto Sessions, which can have common TEK Generation Keys and security parameters. A CSB may also just comprise the MIKEY security policy parameters (see [RFC 3830]).

G.3.2 H.323 domain: Encompasses a single GK zone or a H.323 network among H.323 GK zones.

G.4 Symbols and abbreviations

This annex uses the following abbreviations:

[]	Optional element
{}	Zero, one or more occurrences
a, b, e, d	private DH key of EP A, EP B, GK E, GK D
Cert	digital certificate (see RFC 3830)
CP/C	CallProceeding-to-Connect
CSB	Crypto Session Bundle (see RFC 3830)
CT_B, CT_A	ClearToken for endpoint B, ClearToken for endpoint A (see H.235 Annex I)
DH	Diffie-Hellman
DH_A	DH half-key of endpoint A
DH_B	DH half-key of endpoint B
DRC	Direct-routed Call (see H.235 Annex I)

$ENC_k(x)$	Encryption of x using key k
env_key	Envelope key (RFC 3830) between endpoint B and endpoint A
EP	Endpoint
ESC	H.245 EndSessionCommand
g^a, g^b	Diffie-Hellman half-key of EP A, EP B
g^e, g^d	Diffie-Hellman half-key of GK E, GK D
GK	Gatekeeper
HDR	MIKEY header payload (see RFC 3830)
ID_A, ID_B	Identity (i.e., endpoint ID) of endpoint A, Identity of endpoint B
IETF	Internet Engineering Task Force
Imsg	MIKEY message of the initiator (see RFC 3830)
KEMAC	MIKEY KEMAC payload message (see RFC 3830)
Ma	MIKEY authentication key (see RFC 3830)
MAC(k, x)	Keyed MAC on x using key k
Me	MIKEY encryption key (see RFC 3830)
MIKEY	Multimedia Internet Keying
NTP	Network Time Protocol
PKE	MIKEY PKE payload message (see RFC 3830)
PKI	Public-Key Infrastructure
PRF	Pseudo-Random Function (MIKEY-PRF, see RFC 3830 sections 4.1.2 – 4.1.5)
Rand	random nonce (see RFC 3830)
rand()	random value
Rmsg	MIKEY message of the responder (see RFC 3830)
RSA	Rivest, Shamir and Adleman (public key algorithm)
sa, *sb*	shared secret among endpoint A and GK, shared secret among endpoint B and GK
SDP	Session Description Protocol
SHA1	Secure Hash Algorithm 1 (ISO 10118-3)
SIP	Session Initiation Protocol
sl	shared secret among gatekeepers
SP	Security Policy (see RFC 3830)
SRTCP	Secure Real-time Transport Control Protocol
SRTP	Secure Real-time Transport Protocol (see [RFC 3711])
SSRC	Synchronization source (RTP)
T	Timestamp (see RFC 3830)
TGK	Traffic Generating Key (see RFC 3830) between endpoint A and endpoint B
V	Verification message field (see RFC 3830)
ZZ_{AB}	dynamic shared H.323 secret ZZ_{AB}

G.5 Specification conventions

The object identifiers are referenced through a symbolic reference in the text (e.g., "G1"), clauses G.8.4 and G.9.5 list the actual numeric values for the symbolic object identifiers, for further information see also H.235 clause 5.

Table G.1 defines the five MIKEY key management protocols that are being referenced throughout this annex:

Table G.1/H.235 – MIKEY key management protocols

MIKEY protocol	Description	OID value	Parameter identifier	Implementation
MIKEY	Any MIKEY protocol	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 76}	76	Mandatory to implement
MIKEY-PS	Symmetric key distribution protocol using pre-shared symmetric keys and HMACs, see [RFC 3830].	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 72}	72	Mandatory to implement
MIKEY-DHHMAC	Diffie-Hellman key agreement protocol using pre-shared symmetric keys and HMACs; see [MIKEY-DHHMAC].	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 73}	73	Optional
MIKEY-PK-SIGN	(RSA-based) public-key distribution protocol using digital signatures; see [RFC 3830].	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 74}	74	Mandatory to implement
MIKEY-DH-SIGN	Diffie-Hellman key agreement protocol using digital signatures; see [RFC 3830].	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 75}	75	Optional

MIKEY (see 1st row of Table G.1) refers to the MIKEY protocol family generally without indicating specifically any particular MIKEY key management protocol variant such as MIKEY-PS, MIKEY-DHHMAC, MIKEY-PK-SIGN or MIKEY-DH-SIGN. The related implementation of MIKEY shall encompass processing of the MIKEY messages such as MIKEY common header payload ([RFC 3830] section 6.1), but does not necessarily require any implementation of a particular MIKEY key management protocol or implementation of a particular MIKEY information payload. The corresponding OID and parameter identifier shall be used in those cases where an H.323 endpoint does not know the actually used MIKEY protocol variant. In any other cases, it is recommended to use the specific OID and parameter identifier of the actual MIKEY key management protocol variant instead.

G.6 Introduction

There has been interest in using the security features of IETF SRTP "Secure Real-time Protocol" from within H.235 [H.235]. While former versions of ITU-T Rec. H.235 offer already various media security features such as voice encryption using block ciphers and some limited RTP authentication (anti spam option), there are strong reasons to deploy SRTP:

- use a stream cipher for improved performance, robustness and security;
- be interoperable with other SRTP terminals; such as SIP-based media terminals;

  NOTE – This annex does not specify procedures for security interworking with SIP [RFC 3261]; this is left for further study.
- yield the improved security for RTCP protection;
- obtain improved integrity spanning the entire RTP/RTCP packet;
- deploy state-of-the art AES encryption algorithm;
- use session encryption/authentication keys derived from a pseudo-random function at both ends.

Furthermore, a need has been identified to provide an RSA-based key management in addition to the Diffie-Hellman key agreement schemes provided by ITU-T Rec. H.235. Likewise, non-PKI-based key management techniques are felt useful in the case, where public-key infrastructures are considered not to be the choice. There is also interest to address the issues of lawful interception in the context of key management.

The IETF has also spent efforts to define a real-time capable key management scheme MIKEY [RFC 3830]. This generic key management scheme nicely interfaces with SRTP and is able to provide master keys (TGKs) as well as session traffic keys either end-to-end or possibly end-to-middle/hop-by-hop. MIKEY is an optimized key management protocol that completes within at most two messages, making it ideal for fast start call setup in ITU-T Rec. H.323.

This annex provides security procedures to deploy the MIKEY key management protocols from within H.323/H.235 in order to support SRTP media security. It is noted that there might be other alternative ways by which SRTP could be supported within H.323/H.235 but such measures are not addressed in this annex and remain for further study.

This annex deploys the MIKEY key management protocols in a manner that is conceptually similar to the approach described in [KMGMT-ext] where SIP ([RFC 3261]) carries MIKEY within SDP ([RFC 2327], [SDP-New] and [RFC 3264]).

This annex provides two security profiles with security procedures for two very distinct security infrastructures:

- symmetric key-based security infrastructure supporting multiple gatekeepers (see G.8).
- and asymmetric key-based security infrastructure (PKI) supporting multiple gatekeepers (see G.9).

G.7 Overview and scenarios

Figure G.1 shows the general scenario that this annex addresses. At least two distinct H.323 endpoints A and B are part of this scenario. These endpoints may be H.323 terminals or H.323 media gateways, the latter with potential interface to other packet- or non-packet-based networks. In addition, at least one gatekeeper is assumed to be part of the environment. In case, there is only a single gatekeeper available, it is assumed that all H.323 endpoints are within that single gatekeeper zone only. In case multiple, chained gatekeeper exist, H.323 endpoints may be placed within different gatekeeper zones. It is further assumed that H.323 endpoints communicate directly end-to-end using the RTP media protocol.

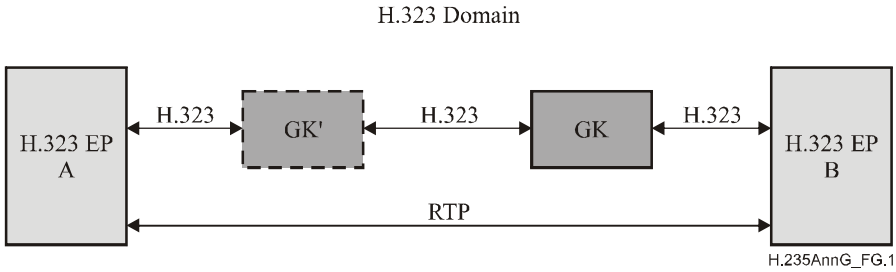

Figure G.1/H.235 – Scenario

Figure G.2 shows the general security scenario indicating the usage of the MIKEY key management protocols and the SRTP media security protocol. The MIKEY key management protocols run between the H.323 endpoints A and B; the MIKEY key management protocols are encapsulated within containers within the H.245 signalling handshakes (Terminal Capability Set, Request Mode, Open Logical Channel handshakes and **MiscellaneousCommand**) and are transparent to any intermediate Gatekeeper(s).

It is noted that an H.323 endpoint may actually be a gateway. For example, such a gateway may provide an interworking function to interface with SIP-based systems. In that case, the gateway may not necessarily terminate MIKEY but may further relay MIKEY and extend MIKEY for truly end-to-end key management among the involved multimedia terminals, supporting thereby end-to-end media security with SRTP. This approach would support security interworking between H.323/H.235 and SIP-based systems. The exact interworking functionality or specification of such gateways is not subject to this annex and is left for further study.

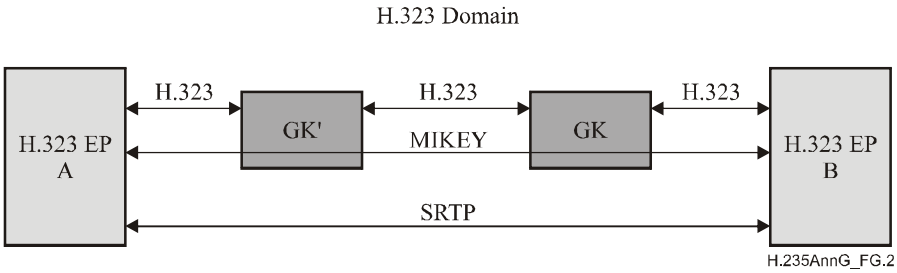

Figure G.2/H.235 – Security scenario with MIKEY and SRTP

All the key management protocols described in this annex proceed in two stages:

- Stage 1 occurs during the H.225.0 RAS and Call Signalling phase. For the symmetric-key MIKEY protocols (MIKEY-PS or MIKEY-DHHMAC), this stage establishes an end-to-end shared ZZ_{AB} between the endpoints A and B that is deployed as a pre-shared secret for MIKEY. For the asymmetric MIKEY protocols (MIKEY-PK-SIGN and MIKEY-DH-SIGN), this stage establishes dynamic shared secrets between the endpoint and its next hop (typically its serving gatekeeper); the dynamic shared secret is not related to MIKEY but serves to secure the H.225.0 Call Signalling between the endpoint and its next hop.
- Stage 2 occurs during the H.225.0 Call Signalling/H.245 protocol phases. This stage negotiates and runs the MIKEY protocol (MIKEY-PS, MIKEY-DHHMAC, MIKEY-PK-SIGN or MIKEY-DH-SIGN) between the endpoints A and B and establishes the MIKEY TGK. During stage 2, the MIKEY endpoints may also run the MIKEY re-keying and key update protocol to refresh or update the TGK. Terminating a call and discarding key material (TGK) may further occur during stage 2.

G.7.1 MIKEY operation at "session level"

The MIKEY key management protocols may operate on a "session-level", i.e., the MIKEY TGK is applied for more than one media stream. It is recommended to run MIKEY at "session level " during the TerminalCapability handshake.

TerminalCapabilitySet shall use **h235SecurityCapability** where **genericH235SecurityCapability** is used within **encryptionAuthenticationAndIntegrity** as follows:

- **capabilityIdentifier** shall hold one of the MIKEY OIDs within **standard**;
- **maxbitRate** and **collapsing** remain unused;
- **nonCollapsing** with the following **GenericParameters** set when MIKEY is executed at a "session level" for all logical channels:
  - **parameterIdentifier**: in **standard** using the value 0 to indicate MIKEY at "session level";
  - **parameterValue** with the MIKEY (I or R) binary-encoded message within **octetString**;
  - **supersedes** remains empty/unused;
- **nonCollapsingRaw** remains unused;
- **transport** (unused or default transport parameters).

OpenLogicalChannel and **OpenLogicalChannelAck** shall not use **encryptionSync** for MIKEY operating at "session level". Likewise, **RequestMode** shall not use **genericModeParameters** of **ModeElement** for MIKEY when MIKEY is operating at "session level".

MiscellaneousCommand shall use **encryptionUpdate** where **genericParameter** is used as follows:

- **parameterIdentifier**: in **standard** using the value of 0 to indicate MIKEY TGK re-keying and CSB updating at "session level";
- **parameterValue** with the MIKEY (I or R) binary-encoded message within **octetString**;
- **supersedes** remains empty/unused.

LogicalChannelNumber shall be ignored for MIKEY at session level and may hold any value.

RequestMode shall use **capabilityIdentifier** within **genericModeParameters** of **ModeElement** as follows:

- **capabilityIdentifier** shall hold one of the MIKEY OIDs within **standard**;
- **maxbitRate** and **collapsing** remain unused;
- **nonCollapsing** with the following **GenericParameters** set when MIKEY is executed at a "session level" for a particular logical channel:
  - **parameterIdentifier**: in **standard** using the value 0 to indicate MIKEY at "session level";
  - **parameterValue** with the MIKEY (I or R) binary encoded message within **octetString**;
  - **supersedes** remains empty/unused;
- **nonCollapsingRaw** remains unused;
- **transport** (unused or default transport parameters).

G.7.2 MIKEY operation at "media level"

Likewise, the MIKEY key management protocols may alternatively operate on a "media-level"; i.e., the MIKEY TGK is applied only for a specific logical channel on a media stream. The

TerminalCapability handshake should be used to negotiate the MIKEY protocol while **OpenLogicalChannel/Ack** should be used to transport the encoded MIKEY message.

TerminalCapabilitySet shall use **h235SecurityCapability** where **genericH235SecurityCapability** is used within **encryptionAuthenticationAndIntegrity** as follows:

- **capabilityIdentifier** shall hold one of the MIKEY OIDs within **standard**;
- **maxbitRate, nonCollapsing** and **collapsing** remain unused;
- **nonCollapsingRaw** remains unused;
- **transport** (unused or default transport parameters).

OpenLogicalChannel or **OpenLogicalChannelAck** shall use the **genericParameter** within **encryptionSync** as follows:

- **parameterIdentifier**: in **standard** using the value of the parameter identifier (see Table G.1) corresponding to the negotiated MIKEY protocol;
- **parameterValue** with the MIKEY (I or R) binary-encoded message within **octetString**;
- **supersedes** remains empty/unused;
- **synchFlag** in **encryptionSync** shall be set to the dynamic payload number. **h235key** shall not be used by this annex and shall be an empty octet string. **escrowentry** shall not be used.

MiscellaneousCommand shall use **encryptionUpdate** where **genericParameter** within **encryptionSync** is used as follows:

- **parameterIdentifier**: in **standard** using the value of the parameter identifier (see Table G.1) corresponding to the negotiated MIKEY protocol;
- **parameterValue** with the MIKEY (I or R) binary-encoded message within **octetString**;
- **supersedes** remains empty/unused.

RequestMode shall use **capabilityIdentifier** within **genericModeParameters** of **ModeElement** as follows:

- **capabilityIdentifier** shall hold one of the MIKEY OIDs within **standard**;
- **maxbitRate** and **collapsing** remain unused;
- **nonCollapsing** with the following **GenericParameters** set when MIKEY is executed at a "media level" for a particular logical channel:
  - **parameterIdentifier**: in **standard** using the value of the parameter identifier (see Table G.1) corresponding to the negotiated MIKEY protocol;
  - **parameterValue** with the MIKEY (I or R) binary-encoded message within **octetString**;
  - **supersedes** remains empty/unused;
- **nonCollapsingRaw** remains unused;
- **transport** (unused or default transport parameters).

G.7.3 MIKEY capability negotiation

If MIKEY protocols are conveyed both in Terminal Capability Set/Request Mode and Open Logical Channel handshake, then the MIKEY in the Open Logical Channel handshake shall take precedence and overwrite prior key management information gained during Terminal Capability Set/Request Mode.

Since endpoints may not implement the full set of all MIKEY key management protocols or may even not have implemented any of them (i.e., endpoints potentially do not support this annex at all), calling endpoints may not know about the supported MIKEY capabilities at the called endpoint.

Therefore, it is recommended that the MIKEY key management capability be negotiated using Terminal Capability Set handshakes.

During terminal capability negotiation, the calling endpoint should indicate its supported and acceptable MIKEY key management protocols. For this, the calling endpoint should indicate its supported MIKEY security capabilities. Within **genericH235SecurityCapability**, the calling endpoint shall set **capabilityIdentifier** to the OID value (see Table G.1) according to the preferred security profile and MIKEY key management**.** The calling endpoint is encouraged to supply also other supported MIKEY protocols, in decreasing preference order according to its security policy and constraints.

A called endpoint that does not support this annex shall reject the call using **ReleaseComplete** with **ReleaseCompleteReason** set to **securityDenied** or may continue unsecured if allowed by its security policy rules. The caller is able to deduce that the callee does not support the requested MIKEY capability by inspecting the returned capability that does not convey a MIKEY capability.

A called endpoint that supports this annex but does not support a requested MIKEY protocol capability shall indicate its supported and acceptable MIKEY protocols during the Terminal Capability Set negotiation handshake.

A called endpoint that supports this annex and a requested MIKEY protocol but does not support a particular combination of MIKEY/SRTP security algorithms and parameters (i.e., MIKEY security policy, SP) shall convey a MIKEY error message as response (see [RFC 3830] sections 5.1.1, 5.1.2 and 6.1.2). The called endpoint should include its supported and acceptable MIKEY security policy (SP) with MIKEY/SRTP security algorithms and parameters.

This annex shall use tunnelling of H.245 messages within H.225.0 Call Signalling for the purpose of securing the H.225.0 Call Signalling messages. This annex may even use non-tunnelling of H.245 message but then it is required that at least an integrity-protected secured transport (TLS, IPsec) be used to secure the H.245 messages. This variant is not further detailed in this annex.

This annex should preferably use fast connect too, where the tunnelled H.245 messages are encapsulated within H.225.0 Call Signalling Setup and CallProceeding-to-Connect. This would allow completing the MIKEY handshakes within two roundtrips at most.

In order to protect against down-grade attacks during capability negotiation, an endpoint conforming to this specification shall adhere to the procedure described in [RFC 3830] section 6.15 where the caller constructs a list of offered MIKEY key management protocol identifiers (KMIDs); see [KMGMT-ext] section 8.3, and includes this list within the MIKEY general extension payload of each offered MIKEY protocol.

For a full-duplex channel, SRTP is instantiated twice, once in each direction; while only one dynamic MIKEY master key (TGK) is negotiated between the H.323 endpoints. The endpoints instantiate directional SRTP session keys by applying distinct MIKEY crypto session identifiers to the MIKEY and SRTP key derivation function.

G.8 Security profile using symmetric security techniques

This clause describes a security profile of this annex where only symmetric security techniques are being deployed.

Figure G.3 shows a scenario that assumes (administered or configured) hop-by-hop shared secrets among the H.323 entities in the H.323 domain (*sa, sb* and *sl*); thereby allowing to deploy H.235 Annex D baseline security (message authentication and/or integrity) of the H.225.0 RAS and Call signalling protocols. For ensuring authenticity (i.e., integrity) of the signalling messages exchanged between EP B and EP A, it is required that H.235 Annex D baseline security is deployed in a hop-to-hop fashion.

Endpoint B is assumed to be loosely time-synchronized with the other H.323 endpoints; otherwise, MIKEY is not able to run securely.

NOTE 1 – This annex does not describe any means how to (securely) synchronize time clocks among the involved entities. It is generally assumed that such time synchronization can be achieved within corporate networks.

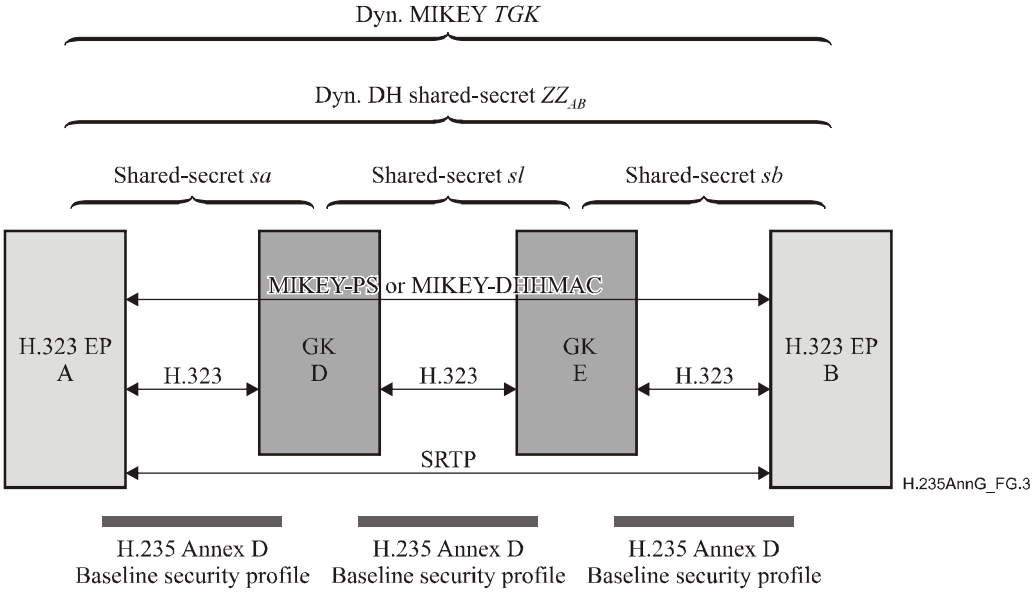

Figure G.3/H.235 – Hop-by-hop scenario only with shared secrets

The basic approach to this scenario is that the MIKEY-PS key distribution protocol (symmetric using pre-shared secrets) or if perfect forward secrecy is of concern the MIKEY-DHHMAC key agreement protocol (Diffie-Hellman using HMAC) is being deployed in the H.323 domain. [MIKEY-DHHMAC] is offered as an option complementing MIKEY, see Appendix G.I.

For EP B (MIKEY initiator) calling EP A (MIKEY responder), a dynamic shared secret ZZ_{AB} is established between EP A and EP B as part of H.225.0 RAS and Setup for a call. The ZZ_{AB} dynamic shared secret is used as the MIKEY pre-shared secret further on, from which MIKEY in EP A and in EP B derives symmetric encryption and authentication keys (not shown in this figure).

The calling EP B generates the MIKEY *TGK* (this is actually a master key) for the peer EP A. EP B builds the MIKEY protocol messages and encapsulates the entire MIKEY message in a container within the tunnelled **TerminalCapabilitySet/OpenLogicalChannel** message. GK E in a GK-routed environment would simply forward the MIKEY container to the other endpoint A without any decoding of MIKEY itself. EP A terminates the MIKEY protocol in the H.323 domain.

Thus, EP B and EP A establish a *TGK*.

The MIKEY-PS or MIKEY-DHHMAC protocol runs between EP B and EP A. In this way, the endpoints obtain the *TGK* and are able to derive the SRTP/SRTCP session keys. SRTP and SRTCP protocols apply these session keys end-to-end.

NOTE 2 – MIKEY provides all the necessary parameters for SRTP (algorithms, key lengths, key lifetime, etc.) as part of the conveyed MIKEY policies.

The gatekeepers are not actively involved in MIKEY processing and act as a store & forwarding relay of the encapsulated MIKEY messages.

For a call setup originating by the EP A, the procedure is similar in the reverse direction with EP A being the initiator and EP B the recipient.

NOTE 3 –

- The scenario shown in Figure G.3 supports also the direct-routed call signalling model with non-routing gatekeeper(s). In such a direct-routed environment, the H.225.0 call signalling messages (Setup, etc.) would be sent end-to-end within the H.323 domain without being relayed by the gatekeeper. See Appendix G.II for illustrations how to use Annex I for that purpose.
- MIKEY uses timestamps within the security protocol as a means to ensure replay protection of the key management message. This demands that the clocks of the endpoints are loosely time-synchronized (within some acceptable clock skew). It is believed that such time-synchronization can be achieved using manually configured time clocks or some network time synchronization protocol (e.g., NTP [RFC 1305]). As such, time synchronization within the H.323 domain should be feasible at least for corporate networks; see also [RFC 3830] sections 5.4 and 9.3
- The combination of fast start and early media in conjunction with the MIKEY-DHHMAC protocol is not recommended. If fast start and early media is required then endpoints shall not use MIKEY-DHHMAC but rather apply MIKEY-PS.
- A scenario with only a single gatekeeper is a special case of the shown scenario with multiple gatekeepers. In this case, far-end gatekeeper/endpoint discovery is not necessary using LRQ/LCF.

The following provides more detailed message flows of the scenario in Figure G.3. This scenario assumes one or more routing gatekeepers within the H.323 domain where H.245 messages are being tunnelled within H.225.0 and fast start is applied.

NOTE 4 – The flow diagrams cover also a direct-routed case (with a non-routing gatekeepers) where H.225.0 call signalling messages are being exchanged directly between the endpoints without being forwarded by any gatekeeper, see Appendix G.II.

The procedure described in this clause establishes an end-to-end shared secret ZZ_{AB} among the H.323 endpoints EP A and EP B during stage 1 using Diffie-Hellman key agreement. This Diffie-Hellman key agreement occurs during the H.225.0 RAS registration and admission phase – and in case of multiple gatekeepers – during the inter-gatekeeper **LRQ/LCF**. The generated Diffie-Hellman shared secret serves as an end-to-end authentication key and lasts during the call. The MIKEY-PS (or MIKEY-DHHMAC) protocol occurs during stage 2 call establishment separately and establishes the MIKEY call-based secrets for the bearer channel.

Appendix G.II describes an alternative and optional procedure using procedure DRC of H.235 Annex I to let the gatekeeper generate and distribute a shared secret to EP A and EP B.

The diagram in Figure G.4 also shows the H.235 Annex D baseline security profile where each message is being secured entirely (authentication and integrity). Yet, similar message flows result when the authentication-only option of the baseline security profile is being applied (not shown). In that case, the HMAC shall not be computed over the entire message but rather only upon a subset (**ClearToken** inside **CryptoToken**) of the RAS/H.225.0 message.

The example message flow shows the case for EP B (MIKEY initiator) calling EP A (MIKEY responder) using fast start (see Figure G.4). The H.323 endpoints A and B initially register with the gatekeeper using **RRQ** and submit their DH half-key (g^a and g^b). The **ClearToken** (within the **CryptoHashedToken**) shall be used to convey the Diffie-Hellman half-key during **RRQ** and **ACF**. For this purpose, **challenge** field shall not be used.

The Diffie-Hellman half-key shall be conveyed in **dhkey** as part of the **ClearToken.** The **ClearToken** shall use OID "TG" (see G.8.5) instead of the baseline Annex D **ClearToken** OID "T", indicating that this security profile is being used in conjunction with H.235 Annex D. The gatekeeper shall keep each half-key as long as the endpoint is registered. Endpoints when executing keep-alives or using lightweight re-registration (re-RRQ) shall not include any DH half-key. The **RCF** shall use the "TG" OID in the **ClearToken** to indicate that the gatekeeper supports this security profile.

EP B trying to call EP A asks for admission at the gatekeeper D (**ARQ**). The **ARQ** shall use the "TG" OID in the **ClearToken**. The OID "TG" shall be used in any other RAS messages within the **ClearToken** too.

The scenario covers multiple, chained gatekeepers but may equally support also only a single gatekeeper. Discovery of the far-end endpoint should be accomplished according to 8.1.6/H.323, "Optional called endpoint signalling" using **LRQ/LCF**. This is how the initiating endpoint locates the far-end GK zone and thereby obtains the Diffie-Hellman half-key of the targeted called endpoint. If GK E needs to locate the far-end GK zone, then GK E shall send a **LRQ** message. For the multicast case, the **generalID** in the CryptoToken of **LRQ** shall not be used. If GK D does not support this profile, then GK D shall return **LRJ**. Otherwise, GK D shall return **LCF** that includes the Diffie-Hellman half-key of EP A. GK E shall then reply with **ACF** including the Diffie-Hellman half-key of EP A. If GK E was not able to locate the far-end endpoint A, then GK E shall return **ARJ**.

The communication between two gatekeepers shall be secured according to H.235 Annex D. For this, it is assumed that a common shared secret *sl* is available. Since **LRQ** among gatekeepers is typically a multicast message, the shared secret *sl* 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. This assumption limits scalability in the general case, and does not provide 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; yet this is left for further study.

EP B obtains the Diffie-Hellman half-key of EP A (**ACF**). The **ACF** shall hold the Diffie-Hellman key of the called endpoint within **dhkey** within the baseline **ClearToken** of Annex D but using OID "TG" instead of "T". Any other fields within the **ClearToken** shall not be modified by this security profile.

NOTE 5 – The endpoints operate with a DH half-key that is static during the overall registration time and for all calls. This should not be a security weakness as long as each endpoint applies truly random half-keys.

However, the endpoints shall provide a fresh 512-bit random value (i.e., 64 octets) within **challenge** along with their DH half-key, see [RFC 2631 section 2.3]. These **challenge** values are call-based and introduce the necessary randomness and timeliness in the DH key generation.

The originating EP B is then able to compute g^{ab} and then the dynamic shared secret ZZ_{AB} using a random **challenge** with the result obtained from MIKEY-PRF(g^{ab} , 0x12F905FE || **challenge**) (see [RFC 3830] sections 4.1.2 – 4.1.5). Then MIKEY is able to derive the encryption (*Me*) and authentication keys (*Ma*) using the MIKEY-PRF (see [RFC 3830] sections 4.1.2 – 4.1.5).

During stage 2, the originating EP B shall generate a fresh MIKEY *TGK* and then shall build the MIKEY I_message Imsg according to the MIKEY-PS protocol using *Me* and *Ma*; also the SRTP session keys can be derived from the TGK as described by [RFC 3711] section 4.3 (not shown in the figures).

The MIKEY I_message shall be binary-encoded.

The originating EP B should always include its DH half-key within **dhkey** in a **ClearToken**, thereby also enabling the GK-supported direct-routed model. The **ClearToken** shall be included as part of the Setup message and shall be sent towards the peer EP A. A routing gatekeeper shall forward the conveyed **ClearToken** (without modification of the MIKEY messages) to the next hop.

The receiving EP A then computes g^{ab} and the dynamic shared secret ZZ_{AB} from MIKEY-PRF(g^{ab}, 0x12F905FE || **challenge**) (see [RFC 3830] sections 4.1.2 – 4.1.5). Then MIKEY derives the encryption (*Me*) and authentication keys (*Ma*) using the MIKEY-PRF (see [RFC 3830] sections 4.1.2 – 4.1.5). Then the conveyed TGKs can be retrieved.

From the *TGK* the receiving EP A is then able to derive the SRTP session keys as described by [RFC 3711] section 4.3 (not shown in the figures).

EP A may build a similar R_message Rmsg but shall build that R_message only when requested by EP B or if necessary (DH). That R_message is being conveyed within the CallProceeding-to-Connect message (CP/C).

The CallProceeding-to-Connect message (CP/C) is sent towards to EP B.

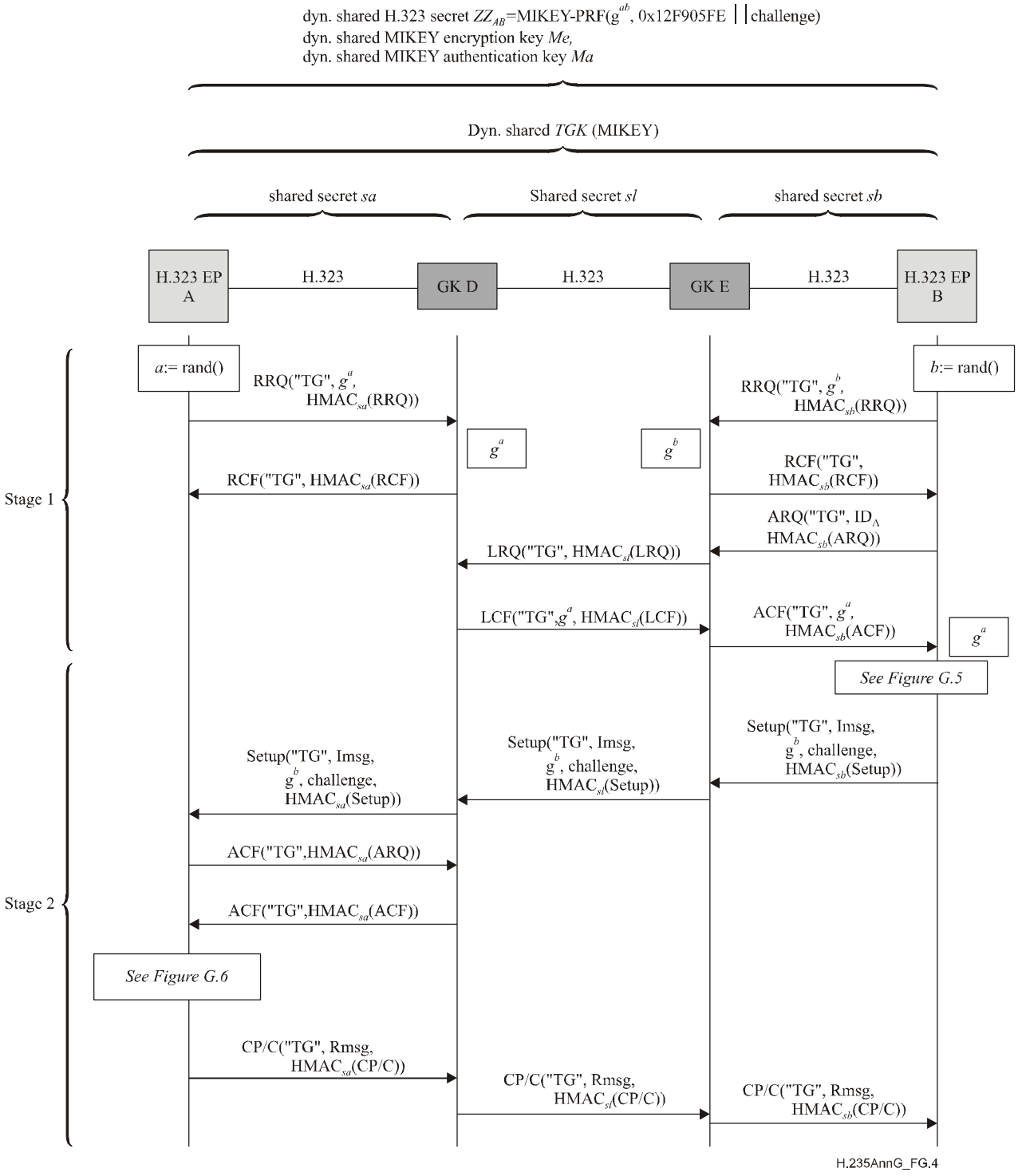

Figure G.4/H.235 – Example Endpoint B calling Endpoint A (GK-routed) with MIKEY-preshared

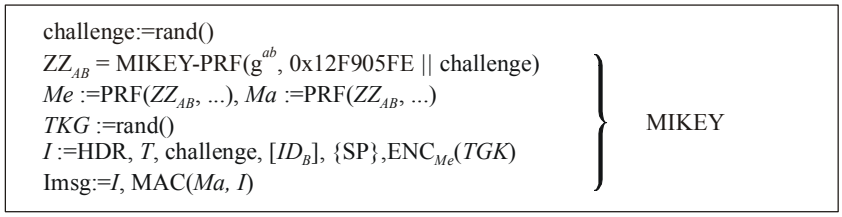

Figure G.5/H.235 – MIKEY-preshared processing by EP B

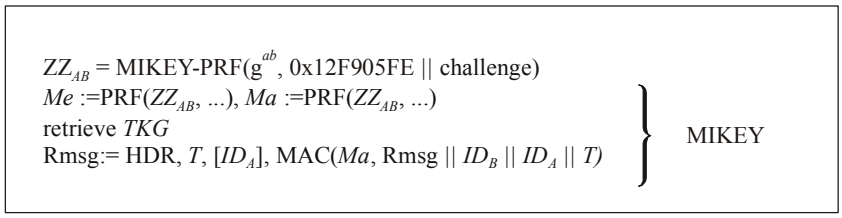

Figure G.6/H.235 – MIKEY-preshared processing by EP A

G.8.1 Terminating a H.323 call

Since the involved endpoints maintain state for MIKEY and SRTP, a proper termination procedure is vital. Figure G.7 shows example message flows in case EP B (MIKEY initiator) terminates a call. Basically, the flow is according to 8.5/H.323, "Phase E – Call termination".

NOTE – The figure shows also optional disengage procedures for the case, when the endpoints completely de-register. Then the endpoints should discard also the private DH-key (*a* or *b*) and the public DH half-key (g^a or g^b).

Since the procedure for terminating a call is independent of this security profile, any applicable OID of the underlying security profile (Annex D, Annex F, etc.) may be used; thus, Figure G.7 does not show any OID.

If the endpoint would register again with the gatekeeper, then new DH half-keys shall be generated. However, complete de-registration is not necessary in any circumstance just for terminating the call. If the endpoint decides to stay registered with the gatekeeper, then the static DH half-keys may continue to be used.

In case the endpoints stay registered and disengage is not being applied, the endpoints shall discard just the call-related information including the peer DH half-key, the **challenge**, the MIKEY keys *Me*, *Ma*, *TGK* and related SRTP session information.

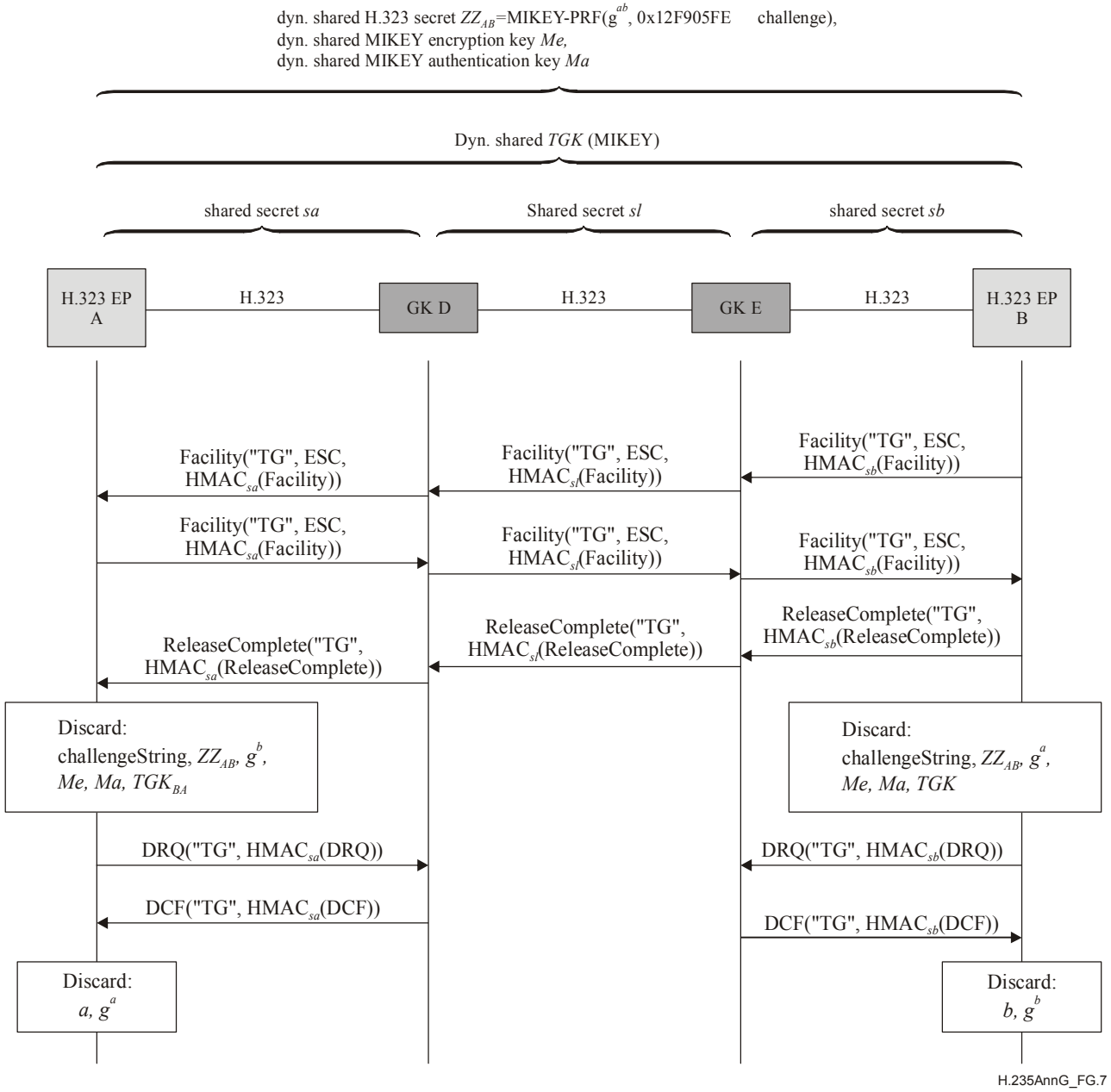

Figure G.7/H.235 – Example Endpoint B terminates a call

G.8.2 TGK re-keying and CSB updating

MIKEY has built-in support for TGK re-keying and/or CSB information updating. The profile of this annex shall use the MIKEY-PS procedure in [RFC 3830] section 4.5 – or if perfect forward secrecy is of concern – the [MIKEY-DHHMAC] section 3.1 for this purpose that allows updating the TGK before expiration or to update other information without changing the TGK.

The TGK re-keying and CSB updating mechanism is useful to protect a bundle of logical channels under the same security policy. For this, it is recommended to run the (full) MIKEY-preshared protocol as described in clause G.8 just for the first logical channel. Any subsequent logical channel that is to apply the same MIKEY security policy or the same TGK should use the CSB updating mechanism without the TGK re-keying mechanism in this clause, by making reference to the initial CSB-ID and by omitting updated TGK data. This allows setting up logical channels or MIKEY crypto sessions more efficiently than by running the full MIKEY protocol on each logical channel.

The MIKEY TGK re-keying or CSB updating messages shall be encapsulated and conveyed in a **MiscellaneousCommand** within a Facility message. The **tokenOID** of the **ClearToken** shall be set to "TG".

If MIKEY is run at "media level", EP B has to determine for which logical channel the TGK re-keying and/or CSB updating should apply. EP A as the responder would equally use the **MiscellaneousCommand** within Facility to convey the MIKEY R_message (if any).

For TGK re-keying (see Figure G.8), the EP B as the MIKEY initiator shall generate a new TGK.

EP A as the responder may confirm the obtained TGK re-keying message, if necessary, as requested by EP B. EP A builds similar R_messages. EP B sends the R_message within the Facility message towards EP A.

For CSB update, the above procedure is similar except that the MIKEY message shall not hold any TGK.

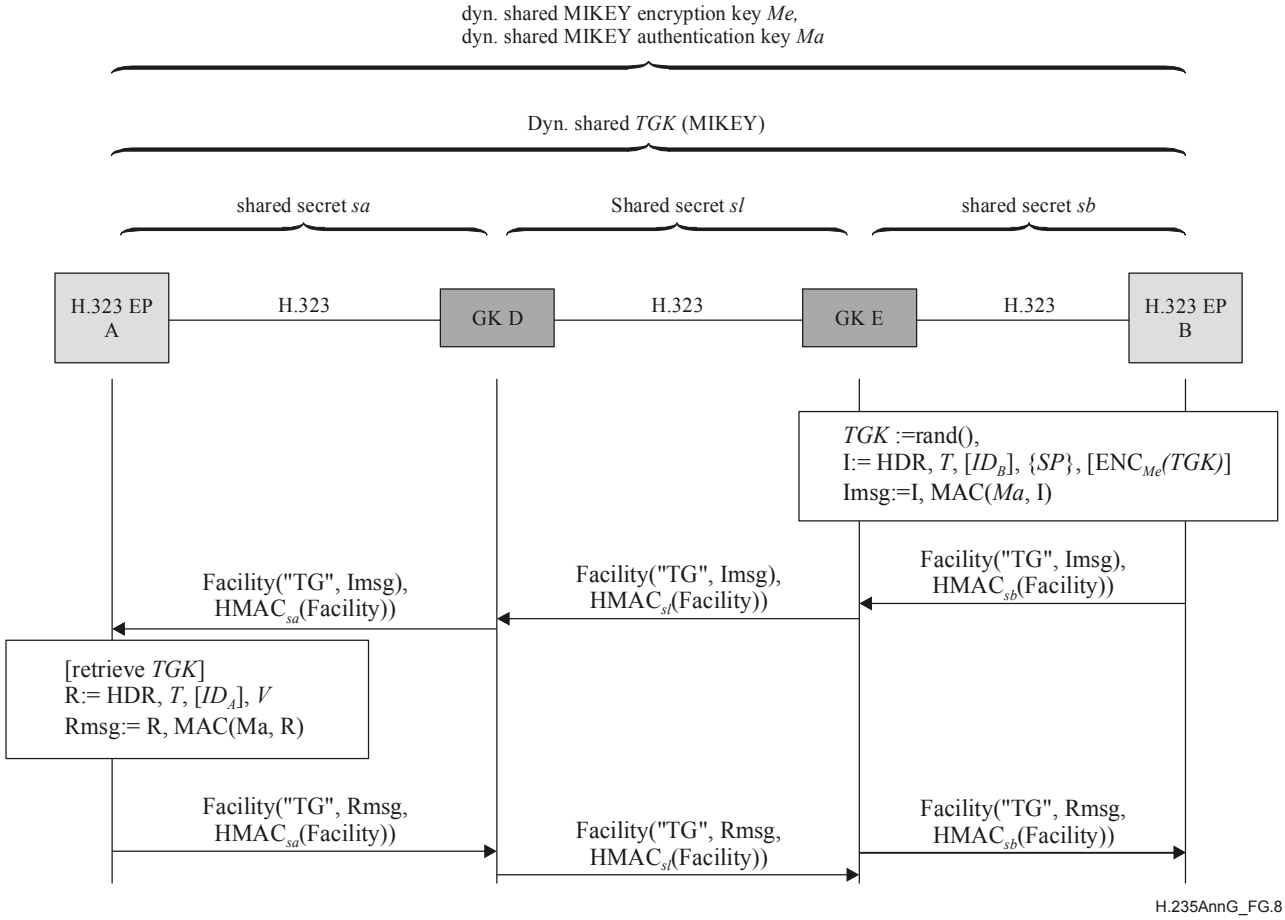

Figure G.8/H.235 – Example Endpoint B updating a key

NOTE – The confirming Facility from EP A to EP B is optional and only necessary when EP B also requested a verification message MIKEY R_message using the V flag in MIKEY HDR.

This annex does not define any procedures for TGK re-keying and/or CSB updating invoked by the responder; this is left for further study.

G.8.3 H.245 tunnelling support

If further logical channels are to be added during a session, H.245 tunnelling mode shall be deployed where the tunnelled H.245 messages are being carried within a Facility message.

G.8.4 SRTP algorithms

This security profile shall use the truncated HMAC-SHA1-32 with an authentication tag length *n_tag* equal to 32 bits as the default authentication algorithm for RTP. Other authentication tag lengths, as those defined by [RFC 3711], shall be supported too and shall be negotiated through the MIKEY security policy (SP) parameter as appropriate.

G.8.5 List of Object Identifiers

"TG"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 70}	Indicates a baseline ClearToken for H.235 Annex D in the context of this annex. This OID also indicates that the MIKEY-PRF is used for computing the shared secret ZZ_{AB}.

G.9 Security profile using asymmetric security techniques

This clause describes a security profile of this annex that deploys asymmetric security techniques. Such a scenario provides most scalability.

The existence of intermediate entities (i.e., gatekeepers) being able to intercept the MIKEY TGK and/or SRTP session keys may not always be acceptable. Figure G.9 shows a scenario that deploys Public-Key Infrastructure (PKI) for establishing SRTP media keys fully end-to-end.

Assumptions: It is assumed that both EP A and EP B possess a private key (*SK*) as well as a certified public key (*cert*). Nevertheless, EP A and GK E as well as EP B and GK D may share (administered/configured) shared secrets in case H.225.0 RAS and call signalling are being secured using H.235 Annex D. It is further assumed that EP A and EP B be loosely time-synchronized, otherwise MIKEY is not able to run securely.

Message authentication/integrity may be achieved either using pre-configured hop-by-hop shared secrets (*sa, sb* and *sl*) and the H.235 baseline security profile, or in a more general case using PKI to establish dynamic shared secrets with the H.235 Annex F Security profile.

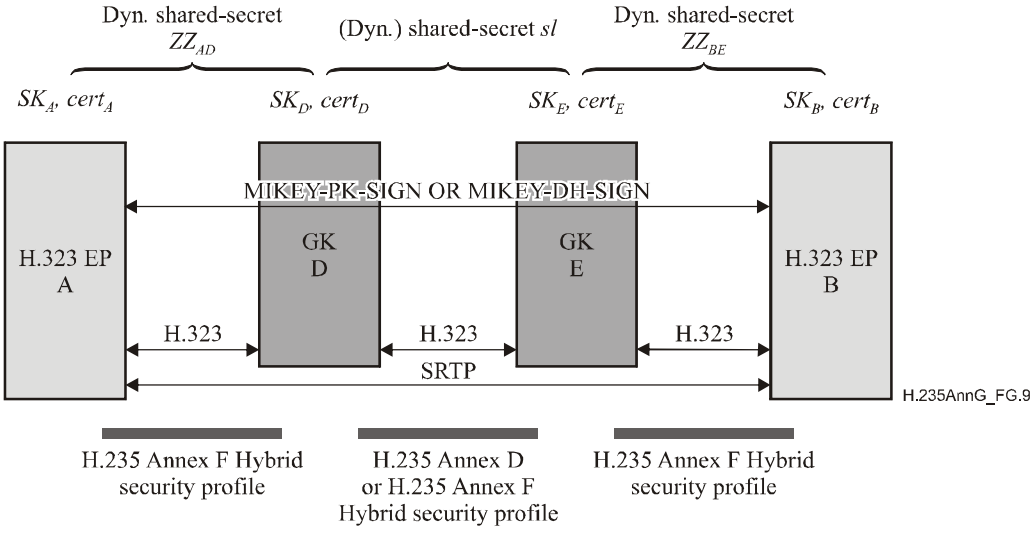

Figure G.9/H.235 – End-to-end scenario using PKI (multiple GKs)

EP A and the EP B run MIKEY-PK-SIGN or MIKEY-DH-SIGN end-to-end and thereby establish the MIKEY TGK from which the end systems derive the SRTP session keys.

NOTE 1 – MIKEY-PK-SIGN satisfies the requirement of an RSA-based key management.

NOTE 2 – Using PKI techniques, the more general H.323 environment featuring multiple, chained gatekeepers in a row should quite certainly be better covered than with less scaleable, and limited architectures using symmetric security techniques.

NOTE 3 – The combination of fast start and early media in conjunction with the MIKEY-DH-SIGN protocol is not recommended. If fast start and early media is required, then endpoints shall not use MIKEY-DH-SIGN but rather apply MIKEY-PK-SIGN.

The following paragraphs provide more detailed message flows of the scenario in Figure G.9. This scenario shows multiple gatekeepers within the H.323 domain.

The following figures further assume a routing gatekeeper (GK-routed model) where H.245 messages are being tunnelled within H.225.0 (Fast start).

NOTE 4 – The flow diagrams cover also a direct-routed case (with a non-routing gatekeeper) where H.225.0 call signalling messages are being exchanged directly between the endpoints without being forwarded by any gatekeeper.

The diagrams also show the H.235 Annex F hybrid security profile where the initial RAS messages are being secured entirely (authentication and integrity) using digital signatures and optional certificates. This is to establish dynamic shared secrets ZZ_{BE} and ZZ_{AD} between the endpoints and the next hop gatekeeper, thereby making static shared secrets superfluous. Yet, similar message flows result when the authentication-only option of the signature security profile is being applied (not shown).

The example message flow shows the case for EP B (MIKEY initiator) calling EP A (MIKEY responder) (see Figure G.10).

During stage 1, H.323 endpoints initially register with the next-hop gatekeeper and submit their DH half-key (g^a and g^b).

EP B tries to call EP A and asks for admission at the gatekeeper E. EP B may query for the peer certificate $cert_c$ by including a security profile element in the **ClearToken** in case certificate information is not yet available to the EP. This security profile element shall use the following fields:

- **elementID** set to 7 indicating a certificate request element; Figure G.10 shows this as certFlag.
- **paramS** remains unused.
- **element** holds an Element where **flag** is set to TRUE.

The ARQ and any following RAS and H.225.0 Call signalling messages are secured by the dynamic shared secret ZZ_{BE} using H.235 Annex D baseline security profile. In case EP B requested certificate lookup, the GK E fetches $cert_c$ from a local or other certificate repository and supplies the result(s) as part of the ACF within **certificate** of the **ClearToken** and includes a security profile element. This security profile element shall use the following fields:

- **elementID** set to 8 indicating a certificate response element; Figure G.10 shows this as certFlag.
- **paramS** remains unused.
- **element** holds an Element where **flag** is set to TRUE.

In case the gatekeeper obtains multiple certificates for a peer endpoint/UA, ACF would actually hold multiple **ClearToken**s – each conveying a single certificate within **certificate**. The endpoint then chooses the appropriate one. However, it may occur that the certificate lookup takes too long; perhaps for example when contacting external repositories. If the gatekeeper is not able to supply the certificate(s) timely or at all, ACF is returned with an empty **certificate** in the **ClearToken** that holds a security profile element where:

- **elementID** set to 8 indicating a certificate response element.

- **paramS** remains unused.
- **element** holds an Element where **flag** is set to FALSE.

It is then the task of the endpoint to abort and attempt to locate the appropriate certificate by means not specified by this annex. In case the gatekeeper would be able to obtain the certificate outside the necessary response time boundary, the gatekeeper should indicate this situation by leaving **certificate** empty and including a security profile element within the **ClearToken** where:

- **elementID** set to 8 indicating a certificate response element.
- **paramS** remains unused.
- **element** holds an Element where **flag** is set to TRUE.

In this case, the GK shall return this **ClearToken** within ACF.

During stage 2, the originating EP B (MIKEY initiator) is then able to generate the fresh MIKEY TGK, and compute the related MIKEY I_message Imsg by applying the MIKEY-PK-SIGN key management protocol (see Figures G.11 and G.12); or if perfect forward secrecy is of concern – the MIKEY-DH-SIGN key agreement protocol (Diffie-Hellman using digital signatures). MIKEY-DH-SIGN is offered as an option.

The SRTP session keys can be derived from the TGK as described by [RFC 3711] section 4.3 (not shown in the figures).

NOTE 5 – Figures G.11 and G.12 do not show every detail of MIKEY, some parts are not shown in the picture.

The MIKEY I_message is binary-encoded and is then being encapsulated in the H.245 **OpenLogicalChannel**.

The **ClearToken** is enclosed as part of the Setup message and is sent towards EP A. A routing gatekeeper forwards the conveyed MIKEY I_message (without modification of the MIKEY message) to the next hop.

In case there are multiple routing gatekeepers, the call signalling messages among the gatekeepers may be secured by applying an administered shared secret and using H.235 Annex D or using H.235 Annex F and private/public keys.

From the *TGK* EP A is then able to derive the SRTP session keys as described by [RFC 3711] section 4.3 (not shown in the figures).

EP A as the MIKEY responder is able to compile the MIKEY R_message Rmsg using the MIKEY *Ma* key and include the MIKEY R_message within the CallProceeding-to-Connect message (CP/C).

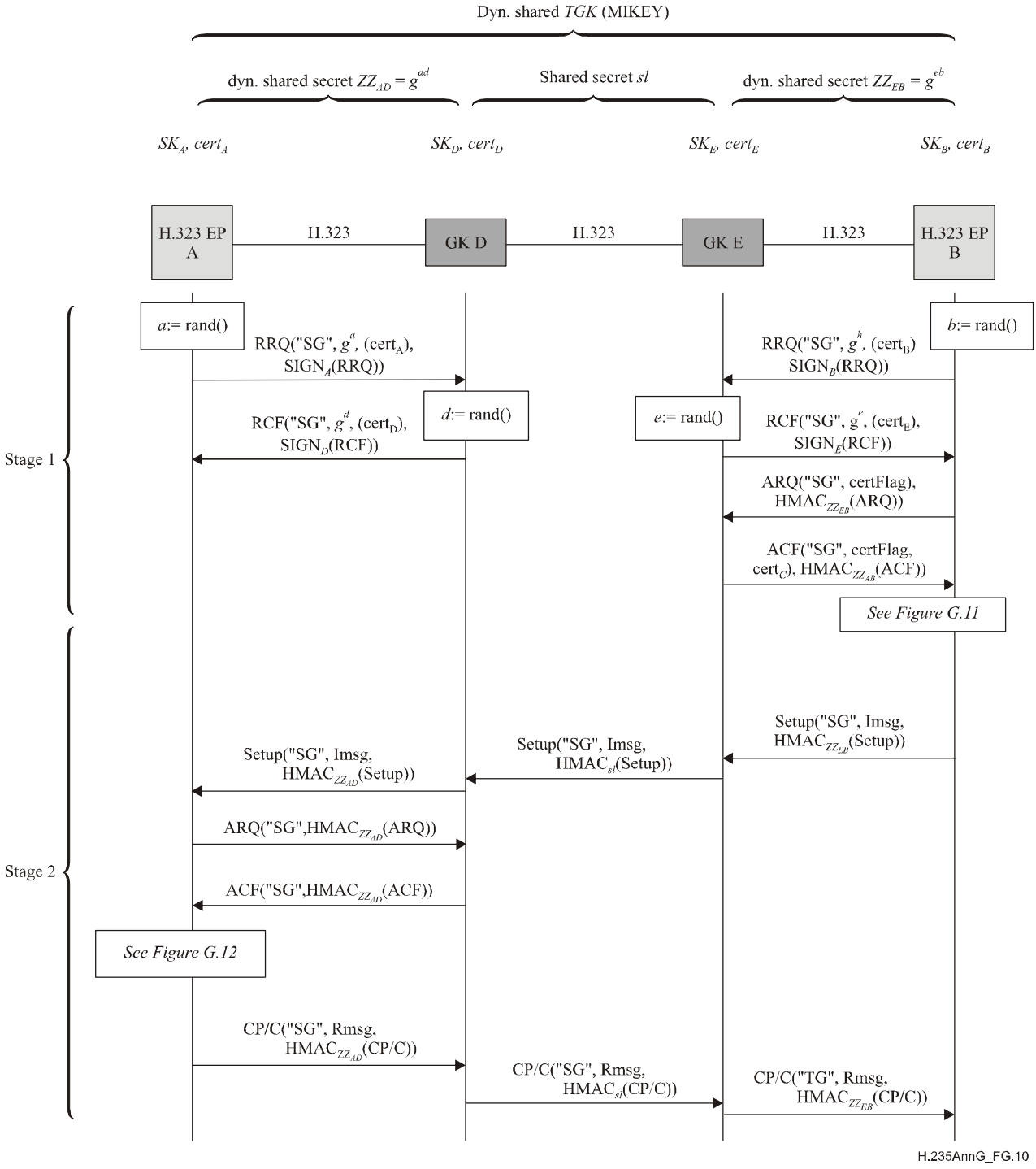

Figure G.10/H.235 – Example EP B calls EP A (multiple GK-routed) with MIKEY-PK-SIGN

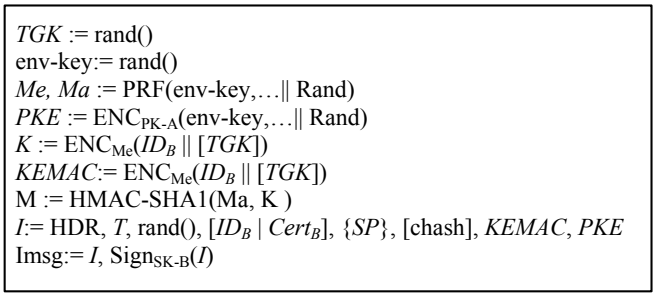

Figure G.11/H.235 – MIKEY-PK-SIGN processing by EP B

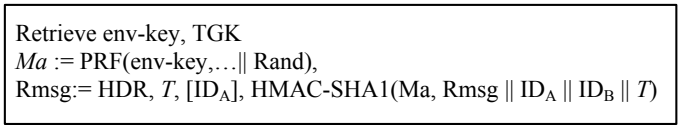

Figure G.12/H.235 – MIKEY-PK-SIGN processing by EP A

A scenario with only a single gatekeeper is a special case of the shown scenario with multiple gatekeepers. In this case, far-end gatekeeper/endpoint discovery is not necessary using LRQ/LCF.

G.9.1 Terminating a H.323 Call

Since the involved endpoints maintain state for MIKEY and SRTP, a proper termination procedure is vital. Figure G.13 shows example message flows in case EP B (MIKEY initiator) terminates a call. Basically, the flow is according to 8.5/H.323, "Phase E – Call termination".

NOTE – The figure shows also optional disengage procedures for the case, when the endpoints completely de-register. Then the endpoints should discard also the private DH-key (*a* or *b*) and the public DH half-key (g^a or g^b).

Since the procedure for terminating a call is independent of this security profile, any applicable OID of the underlying security profile may be used; thus, Figure G.13 does not show any OID.

If the endpoint would register again with the gatekeeper, then new DH half-keys shall be generated. However, complete de-registration is not necessary in any circumstance just for terminating the call. If the endpoint decides to stay registered with the gatekeeper, then the static DH half-keys may continue to be used.

In case the endpoints stay registered and disengage is not being applied, the endpoints shall discard just the call-related information including the peer DH half-key, the **challenge**, the MIKEY keys *Me*, *Ma*, *TGK* and related SRTP session information.

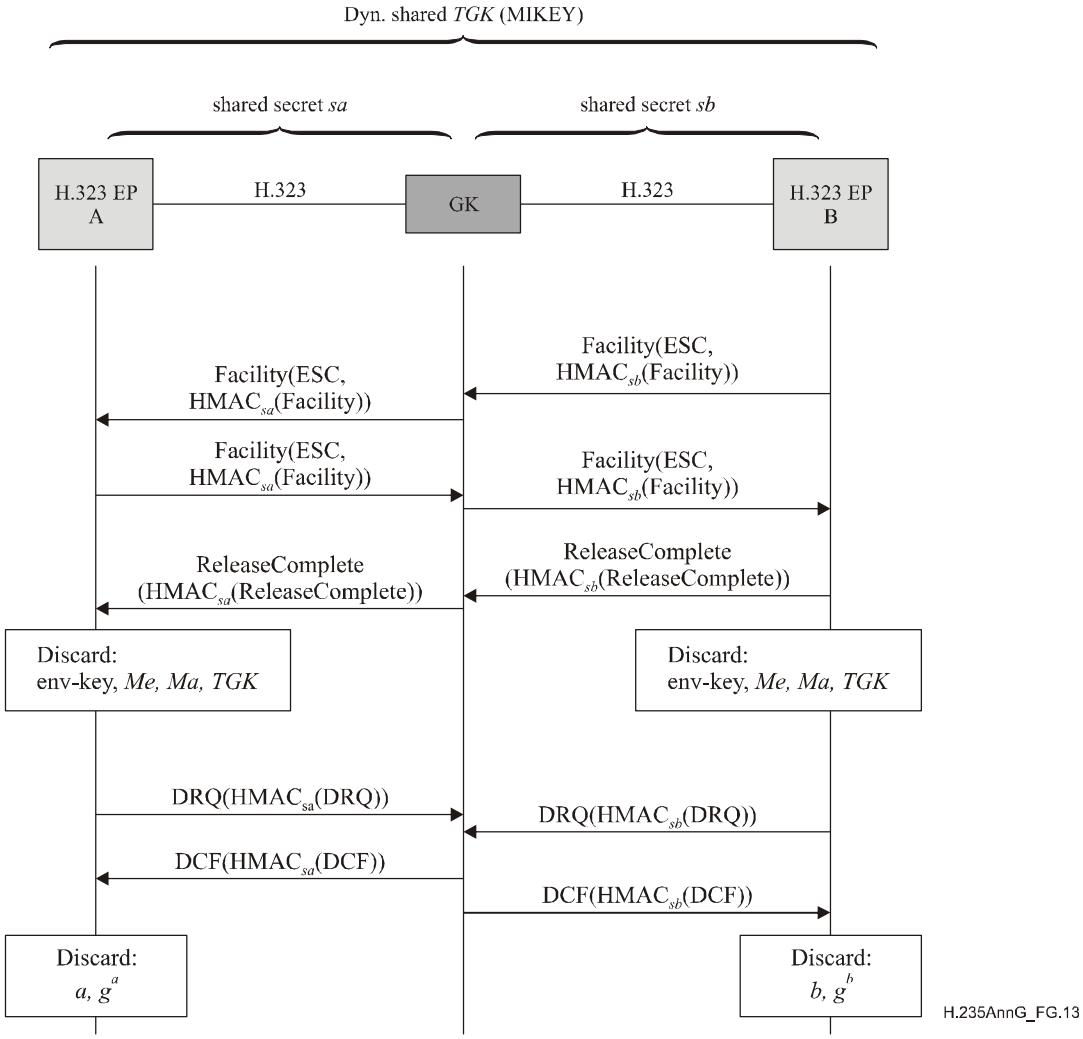

Figure G.13/H.235 – Example Endpoint B terminates a call

G.9.2 TGK re-keying and CSB updating

MIKEY has built-in support for TGK re-keying and/or CSB information updating. This annex shall use the MIKEY-PKSIGN procedure in [RFC 3830] section 4.5 for this purpose that allows updating the TGK before expiration or to update other information (CSB) without changing the TGK.

The TGK re-keying and CSB updating mechanism is useful to protect a bundle of logical channels under the same security policy. For this, it is recommended to run the (full) MIKEY-PKSIGN protocol as described in clause G.8 just for the first logical channel. Any subsequent logical channel that is to apply the same MIKEY security policy or the same TGK, should use the CSB updating mechanism without the TGK re-keying mechanism in this clause by making reference to the initial CSB-ID and by omitting updated TGK data. This allows setting up logical channels or MIKEY crypto sessions more efficiently than by running the full MIKEY protocol on each logical channel.

The MIKEY TGK re-keying or CSB updating messages shall be enclosed within a **MiscellaneousCommand** of a Facility message. The **tokenOID** of the **ClearToken** shall be set to "SG".

For MIKEY at "media level", EP B has to determine for which logical channel the TGK re-keying and/or CSB updating should apply. EP A as the responder would equally use the **MiscellaneousCommand** within Facility to convey the MIKEY R_message (if any).

For TGK re-keying (see Figure G.14), EP B as the MIKEY initiator shall generate a new TGK. **mikey** shall hold the corresponding MIKEY I_message.

The responder (EP A) may confirm the obtained TGK re-keying message if necessary or if so requested by EP B. EP A builds similar R_message Rmsg. This R_message is being conveyed within the Facility message. Rmsg is the corresponding MIKEY response message and shall be conveyed within **octetString** of the **GenericParameter**. EP A sends the Facility message towards to EP B.

For an initiator-initiated CSB update, the above procedure is similar except that the MIKEY message shall not hold any TGK (see Figure G.14).

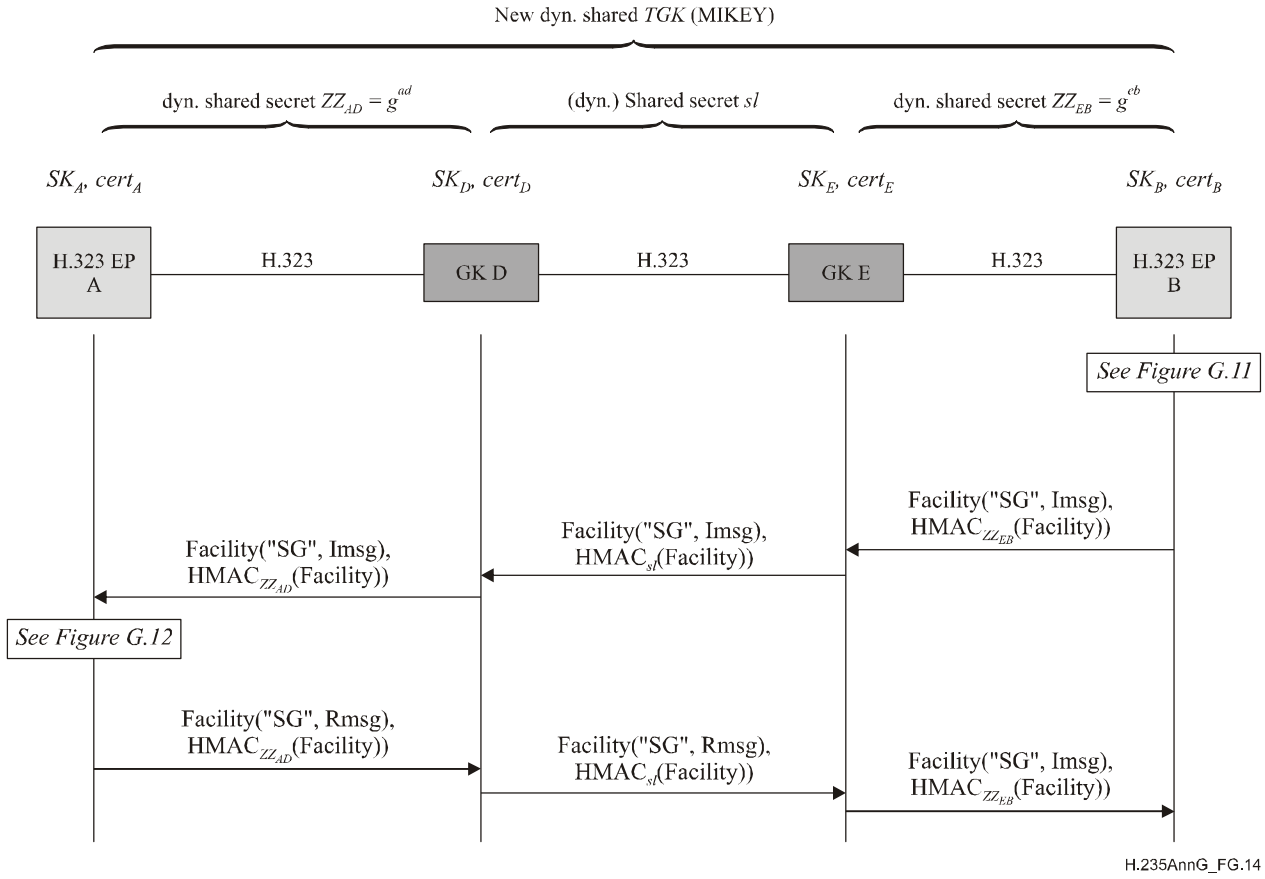

Figure G.14/H.235 – Example EP B (Initiator) initiated TGK re-keying and key update

NOTE – The confirming Facility from EP A to EP B is optional and only necessary when EP B also requested a verification message MIKEY R_message using the V flag in MIKEY HDR.

This annex does not define any procedures for TGK re-keying and/or CSB updating invoked by the responder; this is left for further study.

G.9.3 H.245 tunnelling support

If during a session further logical channels are to be added, H.245 tunnelling mode shall be deployed where the tunnelled H.245 messages are being carried within a Facility message.

G.9.4 SRTP algorithms

This security profile shall use the truncated HMAC-SHA1-32 method with an authentication tag length *n_tag* equal to 32 bits as the default authentication algorithm for RTP. Other authentication tag lengths as those defined by [RFC 3711] shall be supported too and shall be negotiated through the MIKEY security policy (SP) parameter as appropriate.

G.9.5 List of Object Identifiers

"SG"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 71}	Indicates a baseline ClearToken for H.235 Annex F in the context of this annex.

Appendix G.I

MIKEY-DHHMAC option

This informative appendix describes how to deploy the MIKEY-DHHMAC key management option in this security profile.

This key management option assumes only a security infrastructure where shared keys are available. MIKEY-DHHMAC [MIKEY-DHHMAC] provides the security property of perfect-forward secrecy (PFS) due to the inherent capability of the Diffie-Hellman mechanism. Thus, this key management option is applicable when PFS is required and PKI or digital certificates are not available.

This scenario assumes gatekeepers within the H.323 domain.

The procedure described in this clause establishes an end-to-end shared secret among the H.323 endpoints EP A and EP B using a Diffie-Hellman key agreement scheme. This Diffie-Hellman key agreement occurs during the H.225.0 RAS registration and admission phase – and in case of multiple gatekeepers – during the inter-gatekeeper LRQ/LCF. The generated Diffie-Hellman shared secret serves as an end-to-end authentication key and lasts during the call. The MIKEY-DHHMAC protocol occurs during call establishment separately and establishes the MIKEY call-based secrets for the bearer channel.

Figure G.I-1 illustrates an example where endpoint B is calling endpoint A through a routing GK. The flow is similar to Figure G.4 except that the MIKEY-DHHMAC protocol is being deployed. The scenario assumes one or more routing gatekeepers (GK-routed model) where H.245 messages are being tunnelled within H.225.0 (Fast start). Call signalling may or may not pass through a gatekeeper; thus a routing gatekeeper is not necessary to support this scenario.

NOTE 1 – The flow diagram covers also a direct-routed case (with a non-routing gatekeepers) where H.225.0 call signalling messages are being exchanged directly between the endpoints without being forwarded by any gatekeeper.

The diagram in Figure G.I-1 also shows the H.235 Annex D baseline security profile where each message is being secured entirely (authentication and integrity). Yet, similar message flows result when the authentication-only option of the baseline security profile is being applied (not shown). In that case, the HMAC shall not be computed over the entire message but rather only upon a subset (**ClearToken** inside **CryptoToken**) of the RAS/H.225.0 message.

The example message flow shows the case for EP B (MIKEY initiator) calling EP A (MIKEY responder) using fast start (see Figure G.I-1). During stage 1, the H.323 endpoints A and B initially register with the gatekeeper using **RRQ** and submit their DH half-key (g^a and g^b). The **ClearToken** (within the **CryptoHashedToken**) shall be used to convey the Diffie-Hellman half-key during **RRQ** and **ACF**. For this purpose, **challenge** field shall not be used.

The Diffie-Hellman half-key shall be conveyed in **dhkey** as part of the **ClearToken.** The **ClearToken** shall use OID "TG" (see G.8.5) instead of the baseline Annex D **ClearToken** OID "T", indicating that this security profile is being used in conjunction with H.235 Annex D. The gatekeeper shall keep each half-key as long as the endpoint is registered. Endpoints when executing

keep-alives or using lightweight re-registration (re-RRQ) shall not include any DH half-key. The **RCF** shall use the "TG" OID in the **ClearToken** to indicate that the gatekeeper supports this security profile.

EP B trying to call EP A, asks for admission at the gatekeeper D (**ARQ**). The **ARQ** shall use the "TG" OID in the **ClearToken**. The OID "TG" shall be used in any other RAS messages within the **ClearToken** too.

The scenario covers multiple, chained gatekeepers. Discovery of the far-end endpoint should be accomplished according to 8.1.6/H.323, "Optional called endpoint signalling" using **LRQ/LCF**. This is how the initiating endpoint locates the far-end GK zone and thereby obtains the Diffie-Hellman half-key of the targeted called endpoint. If GK E needs to locate the far-end GK zone, then GK E shall send a **LRQ** message. For the multicast case, the generalID in the **CryptoToken** of **LRQ** shall not be used. If GK D does not support this profile then GK D shall return **LRJ**. Otherwise, GK D shall return **LCF** that includes the Diffie-Hellman half-key of EP A. GK E shall then reply with **ACF** including the Diffie-Hellman half-key of EP A. If GK E was not able to locate the far-end endpoint A, then GK E shall return **ARJ**.

The communication between two gatekeepers shall be secured according to H.235 Annex D. For this, it is assumed that a common shared secret *sl* is available. Since **LRQ** among gatekeepers is typically a multicast message, the shared secret *sl* 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. This assumption limits scalability in the general case, and does not provide source authentication. However, it is believed that in corporate networks with a limited, small number of well-known gatekeepers such constraint and security limitations still are acceptable. Securing inter-gatekeeper multicast communication using digital signatures could overcome those limitations; yet this is left for further study.

EP B obtains the Diffie-Hellman half-key of EP A (**ACF**). The **ACF** shall hold the Diffie-Hellman key of the called endpoint within **dhkey** within the baseline **ClearToken** of Annex D but using OID "TG" instead of "T". Any other fields within the **ClearToken** shall not be modified by this security profile.

NOTE 2 – The endpoints operate with a DH half-key that is static during the overall registration time and for all calls. This is not a security weakness as long as each endpoint applies truly random half-keys.

However, the endpoints shall provide a fresh 512-bit random value (i.e., 64 octets) within **challenge** along with their DH half-key, see [RFC 2631 section 2.3]. These **challenge** values are call-based and introduce the necessary randomness and timeliness in the DH key generation.

The originating EP B is then able to compute g^{ab} and then the dynamic shared secret ZZ_{AB} using a random **challenge** with the result obtained from MIKEY-PRF(g^{ab} , 0x12F905FE || **challenge**) (see [RFC 3830] sections 4.1.2 – 4.1.5.). Then MIKEY is able to derive the authentication key (*Ma*) using the MIKEY-PRF (see [RFC 3830] sections 4.1.2 – 4.1.5).

During stage 2, the originating EP B shall generate fresh MIKEY random values *y* with corresponding g^y and then shall build the MIKEY I_message Imsg according to the MIKEY-DHHMAC protocol using *Ma.*

The MIKEY I_message shall be binary-encoded.

The originating EP B should always include its DH half-key within **dhkey** in a **ClearToken**, thereby also enabling the GK-supported direct-routed model. The **ClearToken** shall be included as part of the Setup message and shall be sent towards the peer EP A. A routing gatekeeper shall forward the conveyed **ClearToken** (without modification of the MIKEY messages) to the next hop.

The receiving EP A then computes g^{ab} and the dynamic shared secret ZZ_{AB} from MIKEY-PRF(g^{ab}, 0x12F905FE || **challenge**) (see [RFC 3830] sections 4.1.2 – 4.1.5). Then MIKEY derives the authentication key (*Ma*) using the MIKEY-PRF (see [RFC 3830] sections 4.1.2 – 4.1.5). Then EP A

generates a MIKEY random value *w* and computes g^w. Using the received DH half-keys, EP A computes *TGK*.

From the *TGK* the receiving EP A is then able to derive the SRTP session keys as described by [RFC 3711] section 4.3 (not shown in the figure).

EP A builds similar R_message Rmsg. That R_message is being conveyed within the CallProceeding-to-Connect message (CP/C). Rmsg is the corresponding MIKEY response message sent within a CallProceeding-to-Connect message (CP/C) towards to EP B.

The CallProceeding-to-Connect message (CP/C) is sent towards to EP B.

EP B retrieves the DH half-key and computes *TGK*. EP B then derives the SRTP session keys from the TGKs as described by [RFC 3711] section 4.3 (not shown in the figure).

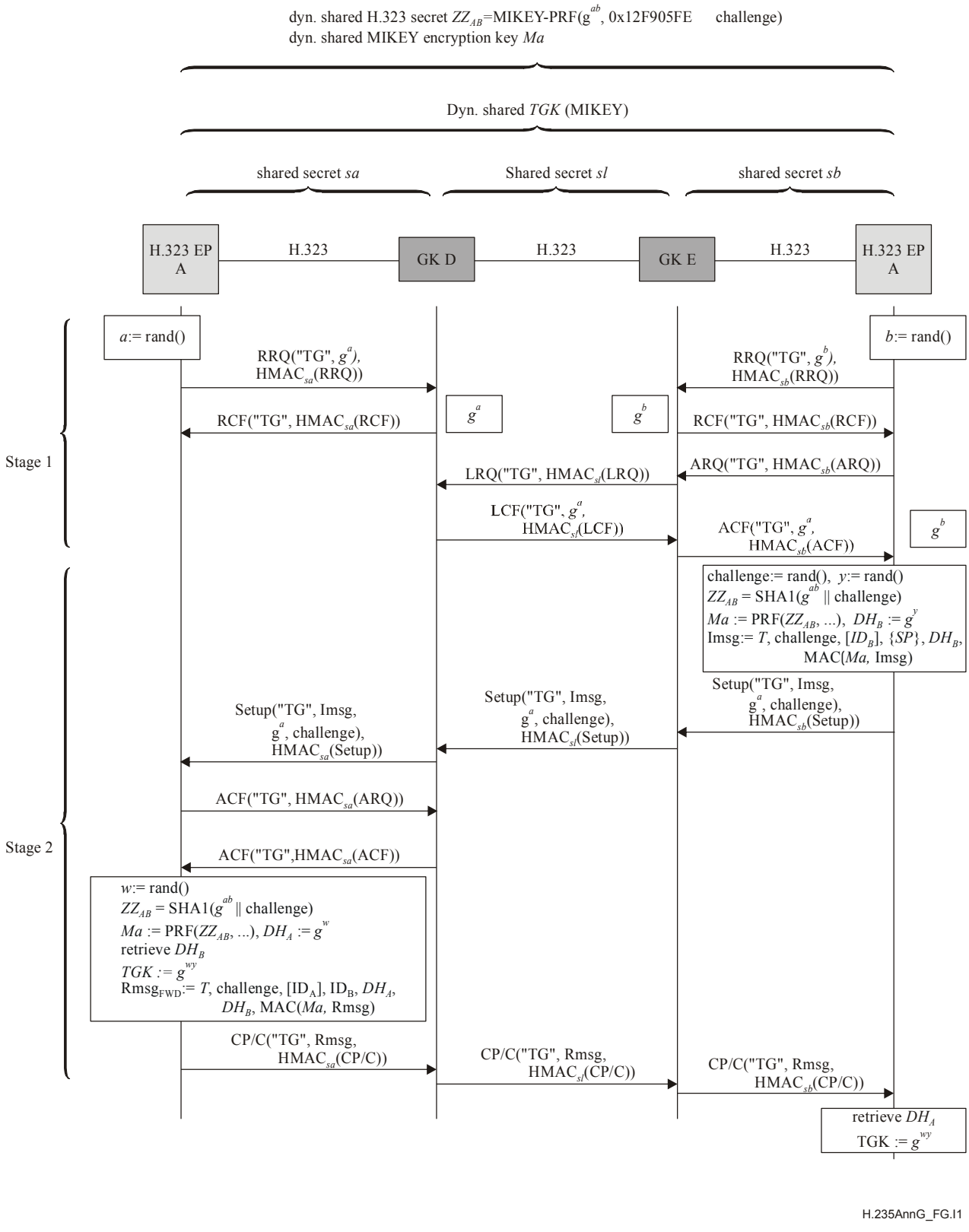

Figure G.I-1/H.235 – Example Endpoint B calling Endpoint A (GK-routed) with MIKEY-DHHMAC

G.I.1 Terminating a H.323 call

Since the involved endpoints maintain state for MIKEY and SRTP, a proper termination procedure is vital. Figure G.I-2 shows example message flows in case EP B (MIKEY initiator) terminates a call. Basically, the flow is according to 8.5/H.323, "Phase E – Call termination".

NOTE – The figure shows also optional disengage procedures for the case, when the endpoints completely de-register. Then the endpoints should discard also the private DH-key (*a* or *b*) and the public DH half-key (g^a or g^b).

Since the procedure for terminating a call is independent of this security profile, any applicable OID of the underlying security profile (Annex D, Annex F, etc.) may be used; thus, Figure G.I-2 does not show any OID.

If the endpoint would register again with the gatekeeper, then new DH half-keys shall be generated. However, complete de-registration is not necessary in any circumstance just for terminating the call. If the endpoint decides to stay registered with the gatekeeper, then the static DH half-keys may continue to be used.

In case the endpoints stay registered and disengage is not being applied, the endpoints shall discard just the call-related information including the peer DH half-key, the **challenge**, the MIKEY keys *Me*, *Ma*, *TGK* and related SRTP session information.

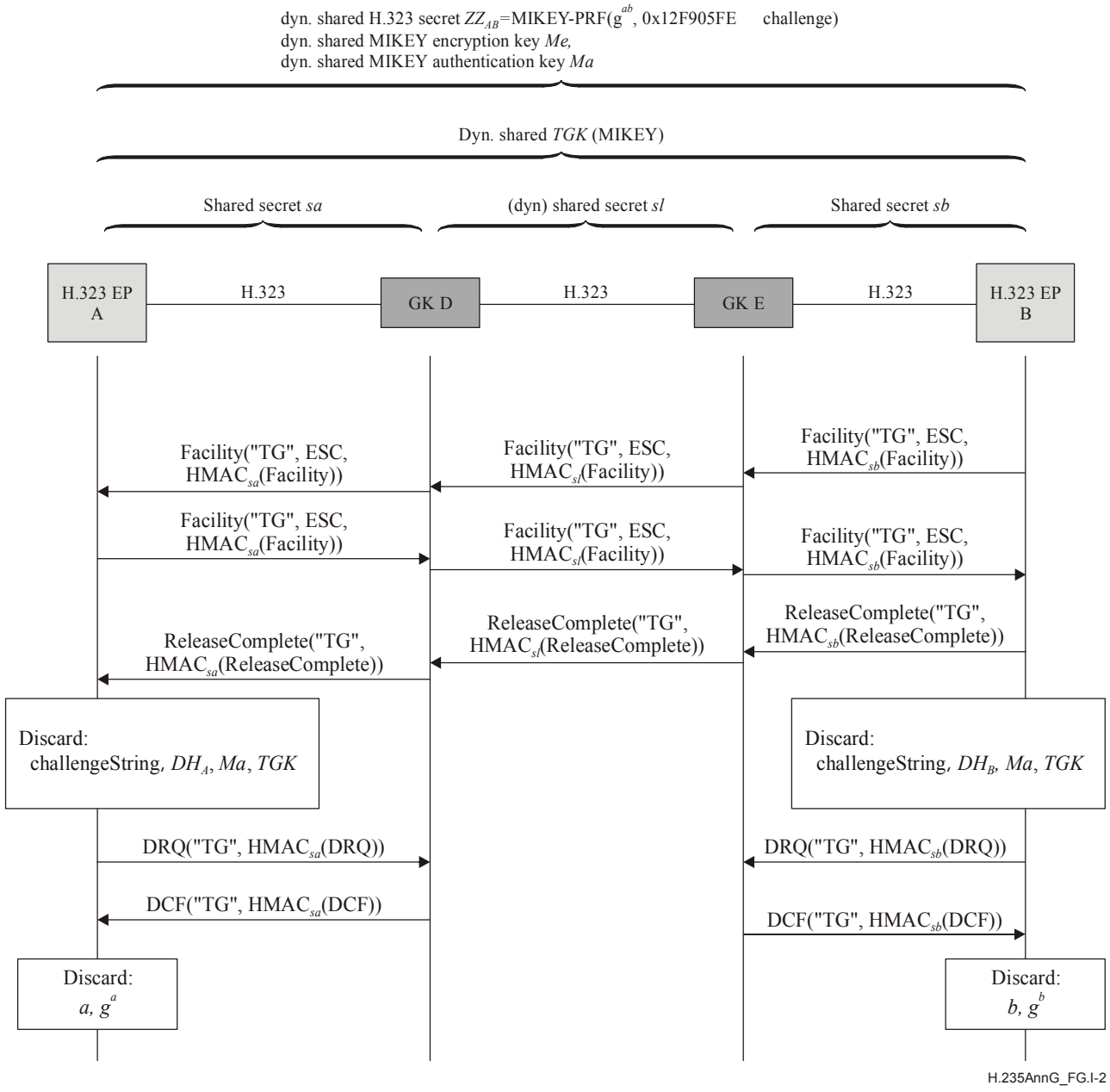

Figure G.I-2/H.235 – Example Endpoint B terminates a call

G.I.2 TGK re-keying and CSB updating

MIKEY has built-in support for TGK re-keying and/or CSB information updating. The profile of this annex shall use the MIKEY-DHHMAC procedure in [MIKEY-DHHMAC] section 3.1 for this purpose that allows updating the TGK before expiration or to update other information without changing the TGK.

The TGK re-keying and CSB updating mechanism is useful to protect a bundle of logical channels under the same security policy. For this, it is recommended to run the (full) MIKEY-DHHMAC protocol as described above just for the first logical channel. Any subsequent logical channel that is to apply the same MIKEY security policy or the same TGK, should use the CSB updating mechanism without the TGK re-keying mechanism in this clause by making reference to the initial CSB-ID and by omitting updated Diffie-Hellman keys. This allows setting up logical channels or MIKEY crypto sessions more efficiently than by running the full MIKEY protocol on each logical channel.

The MIKEY TGK re-keying or CSB updating messages shall be encapsulated and conveyed in a **MiscellaneousCommand** within a Facility message. The **tokenOID** of the **ClearToken** shall be set to "TG".

For MIKEY at "media level", EP B has to determine for which logical channel the TGK re-keying and/or CSB updating should apply. EP A as the responder would equally use the **MiscellaneousCommand** within Facility to convey the MIKEY R_message (if any).

For TGK re-keying (see Figure G.I-3), the EP B as the MIKEY initiator shall generate a new TGK. **parameterValue** shall hold the corresponding binary-encoded MIKEY I_message.

EP A as the responder may confirm the obtained TGK re-keying message if necessary of requested by EP B. EP A builds a similar R_message. This R_message is being conveyed within the Facility message. EP B sends the Facility message towards EP A.

For CSB update, the above procedure is similar except that the MIKEY message shall not hold any TGK.

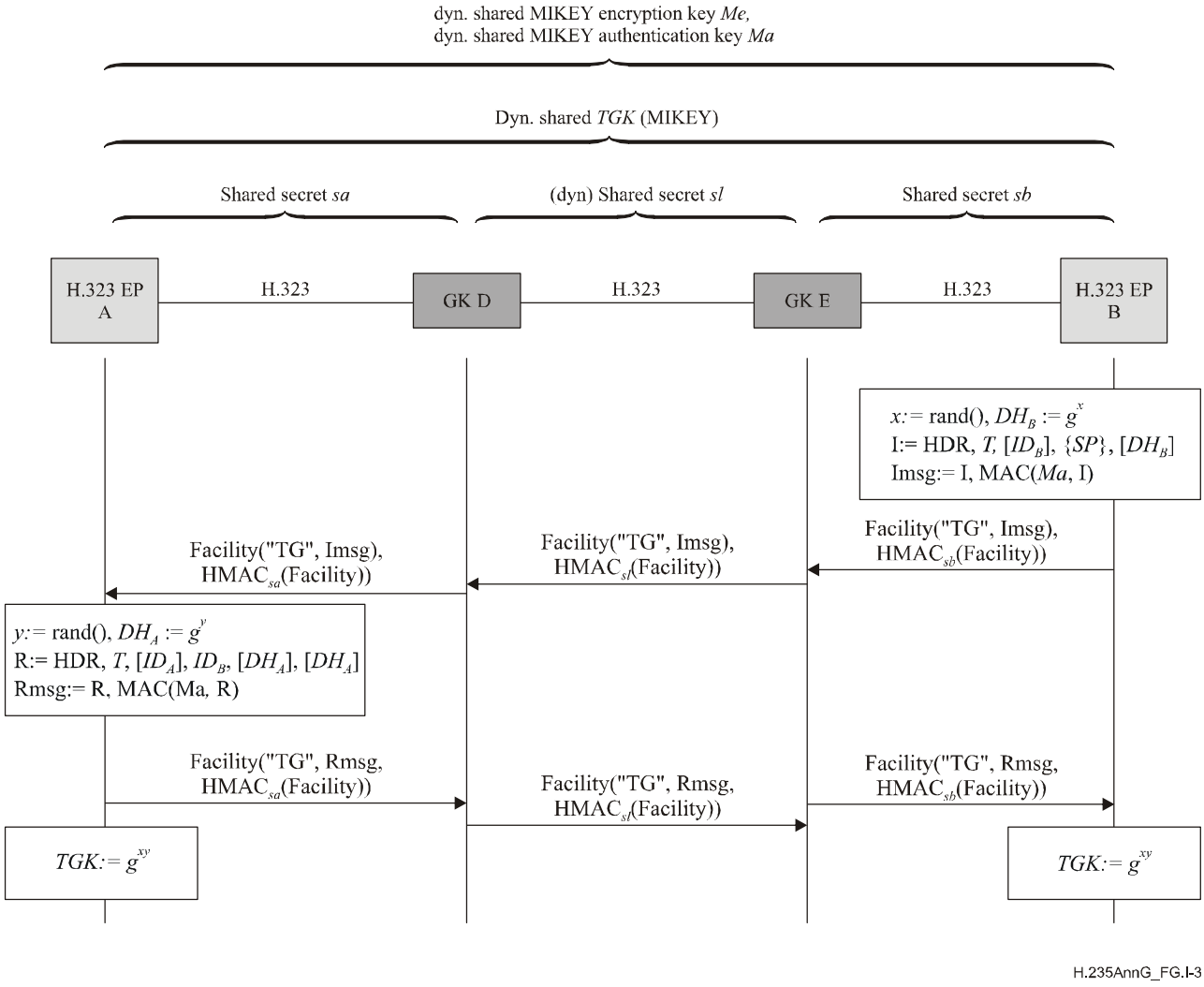

Figure G.I-3/H.235 – Example Endpoint B updating a key

This annex does not define any procedures for TGK re-keying and/or CSB updating invoked by the responder; this is left for further study.

Appendix G.II

Using H.235 Annex I for establishing a pre-shared secret

This informative appendix defines how to deploy Procedure DRC of H.235 Annex I for establishing a pre-shared secret ZZ_{AB} among endpoint B and endpoint A, assuming that no such end-to-end secret exists *a priori*. The method described in this appendix is applicable for the scenario with a single gatekeeper or with multiple gatekeepers. The procedure in this appendix does not involve DH computations during RAS registration or admission but rather deploys symmetric cryptography.

Figure G.II-1 shows the example flow diagram for endpoint B calling endpoint A.

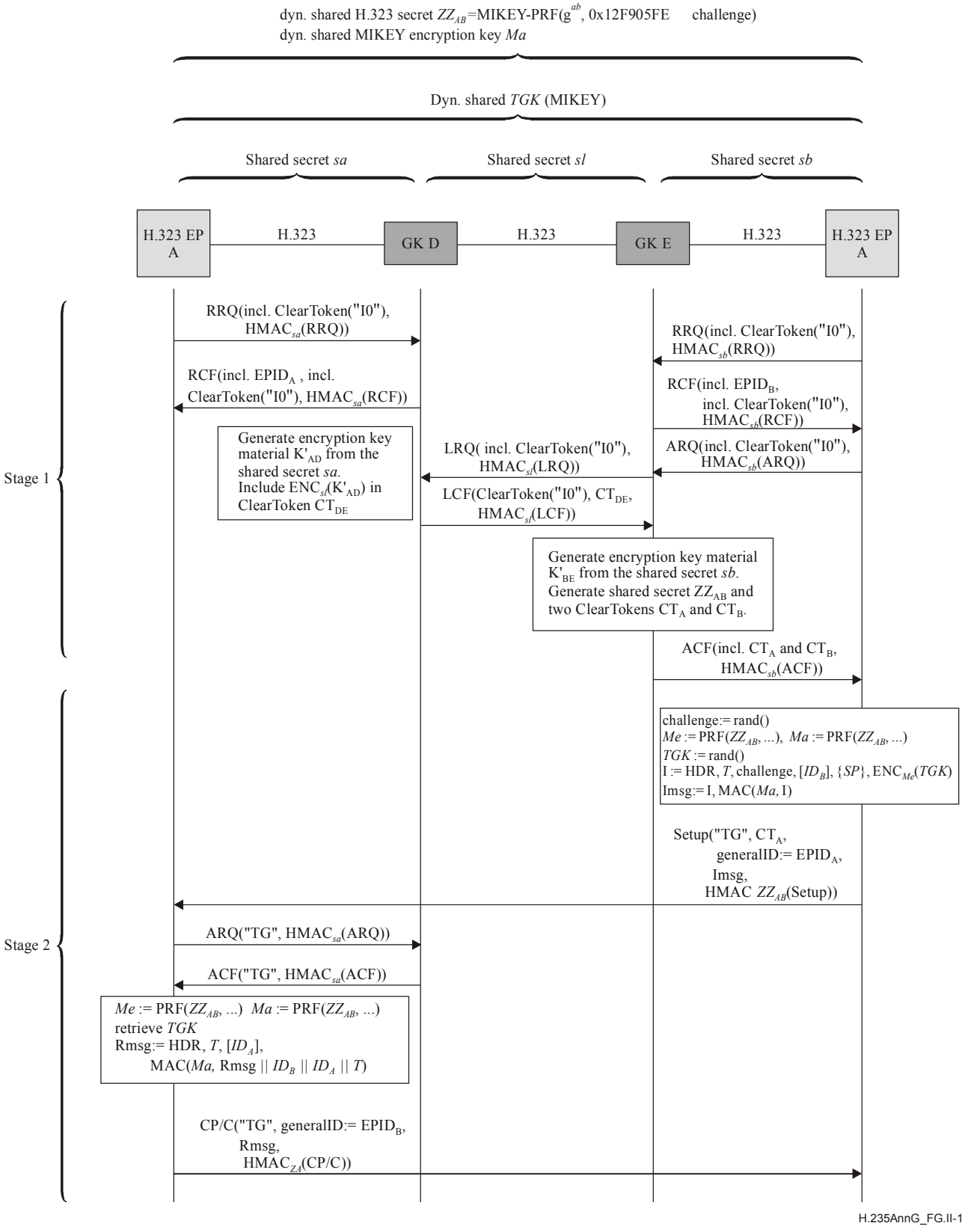

Figure G.II-1/H.235 – Example Endpoint B calling Endpoint A (non-GK-routed) with MIKEY-Preshared and H.235 Annex I DRC

G.II.1 Terminating a H.323 call

The procedure for terminating a H.323 call shall proceed as described in G.8.1.

G.II.2 TGK re-keying and CSB updating

The procedure for TGK re-keying and/or CSB updating shall proceed as described in G.8.2.

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, 2005