International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

H.235.1

(09/2005)

SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS

Infrastructure of audiovisual services – Systems aspects

H.323 security: Baseline security profile

ITU-T Recommendation H.235.1

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.235.1

H.323 security: Baseline security profile

Summary

This Recommendation provides authentication and integrity protection, or authentication-only for H.225.0 RAS and call signalling, H.225.0, and tunnelled H.245 using password-based HMAC-SHA1-96 hash protection of H.225.0 RAS and Call Signalling messages by using secure password-based cryptographic techniques. The security profile is applicable to H.323 terminal-to-gatekeeper, gatekeeper-to-gatekeeper, H.323 gateway-to-gatekeeper and to other H.323 entities in administered environments with symmetric assigned keys/passwords.

In earlier versions of the H.235 subseries, this profile was contained in Annex D/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.235.1 was approved on 13 September 2005 by ITU-T Study Group 16 (2005-2008) under the ITU-T Recommendation A.8 procedure.

Keywords

Authentication, certificate, digital signature, encryption, integrity, key management, multimedia security, security profile.

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.235.1

H.323 Security: Baseline security profile

1 Scope

This Recommendation provides authentication and integrity protection, or authentication-only for H.225.0 RAS and call signalling, H.225.0 and tunnelled H.245 messages using password-based HMAC-SHA1-96 hash protection of H.225.0 RAS and Call Signalling messages by using secure password-based cryptographic techniques. The security profile is applicable to H.323 terminal-to-gatekeeper, gatekeeper-to-gatekeeper, H.323 gateway-to-gatekeeper and to other H.323 entities.

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 version 1 (1998), *Security and encryption for H-series (H.323 and other H.245-based) multimedia terminals*.

– ITU-T Recommendation H.235 version 2 (2000), *Security and encryption for H-series (H.323 and other H.245-based) multimedia terminals*.

– 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.235.2 (2005), *H.323 security: Signature security profile*.

– ITU-T Recommendation H.235.4 (2005), *H.323 security: Direct and selective routed call security*.

– 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.245 version 10 (2003), *Control protocol for multimedia communication*.

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

– ITU-T Recommendation H.323 Annex F (1999), *Simple endpoint types*.

– ITU-T Recommendation Q.931 (1998), *ISDN user-network interface layer 3 specification for basic call control*.

– 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*.

- ITU-T Recommendation X.803 (1994) | ISO/IEC 10745:1995, *Information technology – Open Systems Interconnection – Upper layers security model*.
- ITU-T Recommendation X.810 (1995) | ISO/IEC 10181-1:1996, *Information technology – Open Systems Interconnection – Security frameworks for open systems: Overview*.
- ITU-T Recommendation X.811 (1995) | ISO/IEC 10181-2:1996, *Information technology – Open Systems Interconnection – Security frameworks for open systems: Authentication framework*.
- ISO/IEC 10118-3:2004, *Information Technology – Security techniques – Hash-functions –* Part 3: *Dedicated hash-functions*.

2.2 Informative references

[FIPSPUB180-2] Federal Information Processing Standard FIPS PUB 180-2, Secure Hash Standard, U. S. Department of Commerce, Technology Administration, National Institute of Standards and Technology, 1 August 2002.

[OIW] Stable Implementation – Agreements for Open Systems Interconnection Protocols: Part 12 – OS Security; Output from the December 1994 Open Systems Environment Implementors' Workshop (OIW); http://nemo.ncsl.nist.gov/oiw/agreements/stable/OSI/12s_9412.txt.

[RFC2104] IETF RFC 2104 (1997), *HMAC: Keyed-Hashing for Message Authentication*.

[WEBOIDs] http://www.alvestrand.no/objectid/top.html.

3 Terms and definitions

For the purposes of this Recommendation, the definitions given in clauses 3/H.323, 3/H.225.0 and 3/H.245 apply along with those in this clause. Some of the terms used in this Recommendation are also as defined in ITU-T Recs X.800 | ISO/IEC 7498-2, X.803 | ISO/IEC 10745, X.810 | ISO/IEC 10181-1 and X.811 | ISO/IEC 10181-2.

This Recommendation uses the following terms for provisioning the security services.

3.1 authentication and integrity: This is a combined security service part of the baseline profile that supports message integrity in conjunction with user authentication. The user may ensure authentication by correctly applying a shared secret key procedure. Both security services are provided by the same security mechanism.

3.2 authentication-only: This security service offered by the baseline security profile as an option supports authentication of selected fields only, but does not provide full message integrity. The authentication-only security profile is applicable for signalling messages traversing NAT/firewall devices. The user may ensure authentication by correctly applying a shared secret key procedure.

When using symmetric key techniques, the security services authentication/integrity only apply on a hop-by-hop basis.

4 Symbols and abbreviations

This Recommendation uses the following abbreviations:

ASN.1	Abstract Syntax Notation One
EP	Endpoint
EPID	Endpoint Identifier

GK	Gatekeeper
GKID	Gatekeeper Identifier
GRQ	Gatekeeper Request
HMAC	Hashed Message Authentication Code
ICV	Integrity Check Value
ITU	International Telecommunication Union
LRQ	Location Request
MAC	Message Authentication Code
NAT	Network Address Translation
OID	Object Identifier
RAS	Registration, Admission and Status
RTP	Real-Time Protocol
SHA	Secure Hash Algorithm
TCP	Transmission Control Protocol
UTC	Universal Time Clock
VoIP	Voice over Internet Protocol

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.

This Recommendation defines a **baseline security profile**. The baseline security profile provides basic security by simple means using secure password-based cryptographic techniques. The baseline security profile may be used in conjunction with the security profiles such as H.235.3, H.235.4, H.235.5, H.235.6 and H.235.7.

This Recommendation uses H.235 fields for provisioning authentication/integrity security services upon H.323 signalling messages. Different object identifiers (see clause 15) determine which security service is actually selected and which protocol version of this Recommendation is being used. Procedure I specifies how to implement the security services by certain security mechanisms such as symmetric (keyed hashing) techniques. The object identifiers are referenced through a symbolic reference in the text (e.g., "A"), see also clause 5/H.235.0.

While the message integrity service also always provides message authentication, the reverse is not always true. In practice, combined authentication and integrity service exploit the same key material without introducing a security weakness.

Moreover, all hop-by-hop security information is put into the **CryptoHashedToken** element. This information is re-computed at every hop.

This Recommendation applies certain symmetric cryptographic techniques for the purpose of authentication and integrity. This text uses the term password and shared secret when applying symmetric techniques.

Generally, what password, session key and shared secret all have in common is that they are used in symmetric cryptography among two (or more) entities. The difference between a password and a session key/shared secret is how the keys are actually applied, e.g., passwords for authentication and authorization, session keys for encryption. The term "shared secret" is kind of neutral as it does not actually refer to any specific usage.

The **password** (could be viewed also as a shared secret) is used for the authentication/integrity for RAS and H.225.0, as this item could be entered by the user. A password typically is an alphanumeric character string that users can memorize. The password usually has a longer-term lifetime; the password is known *a priori* and may be defined as part of the overall user subscription process. Some algorithm (e.g., piping the password through a hash algorithm) may transform the password for more convenient processing in the protocols in order to result in a fixed length.

It is obvious that using passwords should be done with care. Passwords are able to provide sufficient security only when they are chosen randomly from a large space, when they convey sufficient entropy such that they are unpredictable and when they are changed periodically. Rules for setting up and maintaining passwords do not fall within the scope of this Recommendation.

A good practice as to how to deploy the benefits from passwords and shared secrets is to transform the user password string into a fixed bit string as the shared secret using a cryptographically strong one-way hash function.

As a recommended example, when using the security profile of this Recommendation, the SHA1 when applied to the password string, yields to a 20-byte shared secret. An advantage is that the hashed result does not only conceal the actual password, but also defines a fixed length bit string format without really sacrificing entropy.

Thus,

shared secret := SHA1 (password)

The H.235 **ClearToken** offers a field called **random** holding a 32-bit integer. This field is used in the following sense: **random** is actually a monotonically increasing number starting at any value and increasing with every outgoing message. The **random** field is used as an additional "randomization" value for input to the keyed-hashed function in the case when several messages are issued shortly one after another, yet convey identical timestamps. This could happen when the UTC clock does not provide sufficient clock resolution. In essence, the produced hash value or integrity check value look different due to the changing **random** value. This is to counter replay attacks. For implementation simplicity, an increasing counter is preferred over a truly random sequence here. The recipient may keep received **timestamp**/**random** pairs during the period defined by a local time window. Replay attacks can be identified when the same **timestamp/random** pair occurs twice.

NOTE – The time window compensates for variances of the synchronized time and for the network transit delay.

This profile defines to "set the **generalID** in the **ClearToken** to the identifier of the recipient". This actually means that, for RAS messages destined for the gatekeeper, this is the GK identifier; for RAS messages destined for the endpoint, this is the endpoint identifier, for H.225.0 call signalling messages destined for the gatekeeper, this is the GK identifier and for H.225.0 call signalling messages destined for the endpoint, this is the called endpoint identifier, see also clause 14.

The **sendersID** shall be set to the identification string of the sender. This actually means that for RAS messages destined for the gatekeeper, this is the endpoint identifier; for RAS messages destined for the endpoint this is the gatekeeper, identifier; for H.225.0 call signalling messages destined for the gatekeeper, this is the GK identifier and for H.225.0 call signalling messages destined for the endpoint, this is the called endpoint identifier, see also clause 14.

This Recommendation may apply message integrity protection that spans the entire message. For H.225.0 RAS, the integrity protection covers the entire RAS message; for call signalling, this covers the entire H.225.0 call signalling message, including the Q.931 headers.

This Recommendation uses well-known security terms such as key, key management and SET, which have different meanings in other contexts (e.g., touch key pad, Q.931/Q.932 feature key management, and Secure Electronic Transaction protocol).

6 Overview

This Recommendation provides authentication and integrity protection, or authentication-only for H.225.0 RAS and call signalling, H.225.0 and tunnelled H.245 messages using password-based HMAC-SHA1-96 hash protection of H.225.0 RAS and Call Signalling messages by using secure password-based cryptographic techniques. The security profile is applicable to H.323 terminal-to-gatekeeper, gatekeeper-to-gatekeeper, H.323 gateway-to-gatekeeper and to other H.323 entities in administered environments with symmetric keys/passwords assigned.

6.1 Summary of security features

The features provided by these profiles include:

– for RAS, H.225.0 and tunnelled H.245 messages:

  - User authentication to a desired entity irrespective of the number of application level hops that the message traverses.

    NOTE – Hop is understood here in the sense of a trusted H.235 network element (e.g., GK, GW, MCU, proxy, firewall). Thus, application level hop-by-hop security, when used with symmetric techniques, does not provide true end-to-end security between terminals.

  - Integrity of the signalling message itself, including the critical portions (fields) of messages arriving at an entity irrespective of the number of application level hops that the message traverses.
  - Application level hop-by-hop signalling message authentication and integrity provides these security services for the entire message.

Several attacks are thwarted by providing the above security services in a suitable fashion. These include:

- Denial-of-service attacks: Rapid checking of cryptographic hash values can prevent such attacks.
- Man-in-the-middle attacks: Application level hop-by-hop message authentication and integrity prevents against such attacks when the man in the middle is between an application level hop, say, a hostile router.
- Replay attacks: Use of timestamps and sequence numbers prevent such attacks.
- Spoofing: User authentication prevents such attacks.
- Connection hijacking: Use of authentication/integrity for each signalling message prevents such attacks.

Other highlights of the simple security profile include:

- Use of robust, well-known and widely deployed algorithms based on IMTC/ETSI/IETF material.
- Capability of deployment in stages based on the security requirement of the business model.
- Applicability to various deployment scenarios such as in closed groups and for scaleable environments and in multipoint conferences.

- The authentication-only security profile is applicable when providing some security for NAT/firewall traversal.

Table 1 summarizes all the procedures defined in this Recommendation by the security profiles dealing with different security requirements. The optional authentication-only security profile is shown as diagonal shading – blue in the electronic copy.

Table 1/H.235.1 – Baseline security profile

Security services	Call functions			
	RAS	**H.225.0**	**H.245 (Note)**	**RTP**
Authentication	Password HMAC-SHA1-96	Password HMAC-SHA1-96	Password HMAC-SHA1-96	
Authentication-only	Password HMAC-SHA1-96	Password HMAC-SHA1-96	Password HMAC-SHA1-96	
Non-repudiation				
Integrity	Password HMAC-SHA1-96	Password HMAC-SHA1-96	Password HMAC-SHA1-96	
Confidentiality				
Access control				
Key management	Subscription-based password assignment			
NOTE – Tunnelled H.245 or embedded H.245 inside H.225.0 fast connect.				

For authentication, the user shall use a password-based scheme. The password-based scheme is highly recommended for authentication due to its simplicity and ease of implementation. Hashing all the fields in the H.225.0 RAS and call signalling messages is the recommended approach for integrity of the messages (also using the password scheme).

Secure H.323 entities with this security profile realize authentication in conjunction with integrity using the same common security mechanism.

Access control means are not explicitly described; they can be implemented locally upon the received information conveyed within H.235 signalling fields (ClearToken, CryptoToken).

This Recommendation does not describe procedures for subscription-based password/secret key assignment with management and administration. Such procedures may take place by means that are beyond the scope of this Recommendation.

The communication entities involved are able to implicitly determine usage of either the baseline security or the signature security profile by evaluating the signalled security object identifiers in the messages (**tokenOID**, and **algorithmOID**; see also clause 15).

6.2 Applicability of the baseline security profile

The baseline security profile is applicable in an environment where subscribed passwords/symmetric keys can be assigned to the secured H.323 entities (terminals) and network elements (GKs, proxies). It provides authentication and integrity, or authentication-only for H.225.0 RAS and call signalling, H.225.0 and tunnelled H.245 using password-based HMAC-SHA1-96 hash as specified by procedure I. H.225.0 call establishment using FastStart (GK-to-GK or terminal-to-terminal) includes integrated key management with Diffie-Hellman.

The baseline security profile mandates the fast connect procedure and recommends to use H.245 tunnelling within H.225.0

6.3 H.323 requirements

H.323 entities that implement this baseline security profile are assumed to support the following H.323 features:

- Fast connect;
- GK-routed model.

6.4 Overview of procedures

The following procedure is described for use in this profile.

Procedure I is a simple symmetric-key-based signalling message authentication mechanism based on a shared password between two entities (e.g., Gatekeeper and H.323 endpoint). This procedure provides authentication and integrity of the RAS, Q.931 and H.245 messages (see clause 7).

Procedure IA is a simple symmetric-key-based authentication-only mechanism based on a shared password between two entities (e.g., Gatekeeper and H.323 endpoint). This procedure provides only authentication but does not provide full message integrity. The authentication-only option is applicable in scenarios where H.323 signalling messages traverse NATs/firewalls.

Depending on the security policy, authentication may be unilateral or mutual by applying the authentication/integrity in the reverse direction as well and thereby providing higher security. The Gatekeeper decides whether to apply authentication/integrity in the reverse direction as well.

Gatekeepers detecting failed authentication and/or failed integrity validation in a RAS or Call signalling message received from a secured endpoint or peer gatekeeper, respond with a corresponding reject message indicating security failure by setting the reject reason to **securityDenial**, or other appropriate security error code, according to 11.1/H.235.0. Depending on the ability to recognize an attack, and the most appropriate way to react to it, a gatekeeper receiving a secured **xRQ** with undefined object identifiers (**tokenOID**, **algorithmOID**) may respond with an unsecured **xRJ** and reject with reason set to **securityDenial**, or it may discard that message. The encountered security event should be logged. On the other hand, the endpoint shall discard the received unsecured message, time out and may retry once again by considering to choose different OIDs. Likewise, a gatekeeper receiving a secured H.225.0 SETUP message with undefined object identifiers (**tokenOID**, **algorithmOID**) may respond with an unsecured RELEASE COMPLETE and reject reason set to **securityDenied**, or may discard that message. Similarly, the encountered security event should be logged.

There is implicit H.235 signalling for indicating use of procedure I and the applied security mechanism, based upon the value of the object identifiers (see also clause 15) and the message fields filled in.

This profile does not use the H.235 ICV fields; rather cryptographic integrity check values are treated as cryptographic hash values and are put into the hash fields of the **CryptoToken**.

7 Symmetric-key-based signalling message authentication and integrity (procedure I)

The procedures below shall be followed when procedure I is employed:

- The HMAC-SHA1-96 algorithm generates a 12-byte (96-bit) hash value as the resulting authenticator. If the key is generated from a password, the mechanism described in 8.2.4/H.235.0 *shall* be used for computing the key from the password.

  NOTE 1 – When the secret key is derived from a user-entered password, care should be taken to ensure sufficient randomness. It is recommended, for example, to use truly random secrets for the secret key, or to ensure that random passwords are sufficiently long.

- The **CryptoH323Token** field in each RAS/H.225.0 message shall contain the following fields:

- **nestedCryptoToken** containing a **CryptoToken** which itself contains the **cryptoHashedToken** containing the following fields:
  - **tokenOID** set to "A", indicating that the authentication/integrity computation includes all fields in the H.225.0 RAS and call signalling message.
  - **hashedVals** containing the **ClearToken** field used with the following fields:
    - **tokenOID** set to "T", indicating that the baseline **ClearToken** as shown below is being used for message authentication and replay protection and optionally also for Diffie-Hellman key management as described in 8.5/H.235.6. Alternatively, other ClearTokens with other OIDs may be used in place of the baseline ClearToken.
    - **timeStamp** contains the timestamp.
    - **random** contains a monotonically increasing sequence number. This number allows the construction of two messages with the same timestamp (within the clock resolution).
    - **generalID** contains the identifier of the recipient (only in case of unicast messages).
    - **sendersID** contains the identifier of the sender.
    - **dhkey**, used to pass the Diffie-Hellman parameters as specified in this Recommendation during **Setup** to **Connect**.
      - **halfkey** contains the random public key of one party.
      - **modsize** contains the DH-prime (see Table 4/H.235.6).
      - **generator** contains the DH-group (see Table 4/H.235.6).

    NOTE 2 – When the baseline security profile is used without the voice encryption security profile, then no Diffie-Hellman parameters should be sent and **dhkey** should be absent; **halfkey**, **modsize** and **generator** may be set to {'0'B,'0'B,'0'B}.
- **token** containing **HASHED** with the fields:
  - **algorithmOID** set to "U" indicating the use of HMAC-SHA1-96.
  - **params** set to NULL.
  - **hash** containing the authenticator computed using HMAC-SHA1-96. The authenticator can be computed over:
    - all the H.225.0 RAS and call signalling fields of the message if **tokenOID** in the **CryptoHashedToken** is set to "A" (indicating authentication and integrity).

  tokenOID "A" is used for protection of tunnelled H323-UU-PDUs including all H.245 message contents; the hash computation shall be done over the entire **H.225.0** call signalling message with all fields according to the procedure described in 7.3.

- The authenticator is verified at the end of each channel terminating leg (EP1-GK1, GK1-GK2, GK2-EP2, EP1-GK2, GK1-EP2 or EP1-EP2 as the case may be), and recomputed prior to sending the message out on the subsequent leg.

NOTE 3 – The authenticator is computed on a per-message basis.

NOTE 4 – The padding method within the SHA1 standard (ISO/IEC 10118-3) shall be used.

NOTE 5 – When the combined authentication and integrity is being used, the authenticator is computed over the entire message.

NOTE 6 – In order to prevent the possibility of replay attacks, it is highly recommended that implementations ensure that the password (key) is changed prior to a turn-around (or cycle completion) of the monotonically increasing sequence number.

NOTE 7 – The recipient is able to detect usage of procedure I by evaluating the **tokenOID** within the hashed **EncodedGeneralToken** (detecting presence of "A").

7.1 Computation of the password-based hash

Both sender and receiver of an authenticated/integrity protected message compute a keyed hash over all the ASN.1-coded message fields (using OID "A"). For the authentication-only profile, both sender and receiver compute a keyed hash over all the ASN.1 coded ClearToken (using OID "B").

7.2 HMAC-SHA1-96

HMAC-SHA1-96 is the truncated 96-bit cryptographic hash value of the 160-bit SHA1 computation. The 96 leftmost bits of the network byte order representation of the hash value shall be used as the result. RFC 2104 describes the procedure with the secret key *K* set to the shared secret (= SHA1-hashed password) and *text* set to the message buffer.

7.3 Computation and verification of authentication and integrity

For authentication and message integrity (in case OID "A" is applied), the procedure is as follows.

The sender of a message shall compute the hash as follows:

1) Set the hash value to a specific default pattern with a length of 96 bits. The exact bit pattern does not matter, but a good choice is a unique bit pattern that does not occur in the remaining message.
2) ASN.1 – encode the entire message; for RAS this shall include the entire H.225.0 RAS message; for call signalling, this shall include the entire H.225.0 call signalling message.
3) Locate the default pattern in the encoded message; overwrite the found bit pattern all with 96 zero bits.

   NOTE 1 – This location may involve some trial-and-error steps in the rare case when the default pattern occurs more than once in the message.
4) Compute the cryptographic hash value upon the ASN.1 encoded message using HMAC-SHA1-96 (see 7.2).
5) Substitute the default pattern in the encoded message with the computed hash value.

The recipient receives the message and then proceeds as follows:

1) ASN.1 – decode the message.
2) Extract the received hash value and keep it in a local variable RV.
3) Search and locate the hash value RV in the received encoded message.

   NOTE 2 – In rare circumstances where the hash value substring might occur several times in the entire message, steps 3-6 have to be iterated successively with a different starting search position.
4) Overwrite the bit pattern in the encoded message all with 96 zeros.
5) Compute the cryptographic hash value upon the encoded message using HMAC-SHA1-96 (see clause 7.2).
6) Compare RV with the computed hash value. The message is considered uncorrupted only if both hash values are equal; in this case, the authentication is successful and the procedure stops.
7) Otherwise, repeat steps 3-7 by restoring RV to the previous location and search for another match. If none of the matches yield a correct hash value comparison, then the authentication has failed and the message has been altered (accidentally or intentionally) during transit.

8 Authentication-only (procedure IA)

Terminals may choose to implement authentication-only (using OID "B", see clause 20/H.235.2). In this case, the authenticator is computed just over a subset (**ClearToken** inside **CryptoToken**) of the RAS/H.225.0 message. Authentication-only may be useful for traversing NAT/firewalls that change IP addresses/ports within the H.323 payloads.

Since the authentication spans only a very limited portion of the message, authentication-only does not provide message integrity as procedure I features. Thus, authentication-only provides less security.

For authentication-only, the following fields shall be used in the protected messages:

- The **CryptoH323Token** field in each RAS/H.225.0 message shall contain the following fields:
  - **nestedCryptoToken** containing a **CryptoToken** which itself contains the **cryptoHashedToken** containing the following fields:
    - **tokenOID** set to:
      - "B" (see clause 20/H.235.2) indicating that the authentication-only computation includes all fields in the **ClearToken**.
    - **hashedVals** containing the **ClearToken** field used with the following fields:
      - **tokenOID** set to:
        - "T" (as the baseline ClearToken example for the remainder of ClearToken contents) or any suitable OID for any other purposes.
      - **timeStamp** contains the timestamp;
      - **random** contains a monotonically increasing sequence number. This number allows making two messages with the same timestamp (within the clock resolution) unique;
      - **generalID** contains the identifier of the recipient (only in case of unicast messages);
      - **sendersID** contains the identifier of the sender;
      - **dhkey**, used to pass the Diffie-Hellman parameters as specified in ITU-T Rec. H.235.0 during **Setup** to **Connect**.
        - **halfkey** contains the random public key of one party;
        - **modsize** contains the DH-prime (see Table 4/H.235.6);
        - **generator** contains the DH-group (see Table 4/H.235.6).

        NOTE 1 – When the baseline security profile is used without the voice encryption security profile, then no Diffie-Hellman parameters should be sent and **dhkey** should be absent; **halfkey**, **modsize** and **generator** may be set to {'0'B,'0'B,'0'B}.
  - **token** containing **HASHED** with the fields:
    - **algorithmOID** set to "U" indicating the use of HMAC-SHA1-96;
    - **params** set to NULL;
    - **hash** containing the authenticator computed using HMAC-SHA1-96. The authenticator shall be computed over:
      - all the fields of the **ClearToken** if **tokenOID** in the **CryptoHashedToken** is set to "B" (indicating authentication-only).
- The authenticator is verified at the end of each channel terminating leg (EP1-GK1, GK1-GK2, GK2-EP2, EP1-GK2, GK1-EP2 or EP1-EP2 as the case may be), and recomputed prior to sending the message out on the subsequent leg.

NOTE 2 – The authenticator is computed just on the **ClearToken**.

NOTE 3 – The padding method within the SHA1 standard (ISO/IEC 10118-3) shall be used.

NOTE 4 – In order to prevent the possibility of replay attacks, it is highly recommended that implementations ensure that the password (key) is changed prior to a turn-around (or cycle completion) of the monotonically increasing sequence number.

NOTE 5 – The recipient is able to detect usage of procedure IA by evaluating the **OID** "B" within the **tokenOID**.

The authenticator shall be computed just over the **ClearToken** inside the **CryptoH323Token** (i.e., **ClearToken**) of the **token** of the **cryptoHashedToken**. The cryptographic hash shall be computed over the ASN.1 encoded bitstring of **ClearToken**.

H.235 version 1 and version 2 endpoints may use authentication-only, in which case the corresponding OIDs for "B" shall be used. H.235 version 1 endpoints shall adhere to the procedure described in clause 11.

9 Usage illustration for procedure I

Figures 1 through 3 depict the presence of shared keys at the end of communicating channels for the different combinations of gatekeeper and direct-routed H.225.0 channels. Irrespective of the call model, a secret key is always present between an EP and its GK in order to provide for RAS message authentication and integrity. When a RAS channel and an H.225.0 channel terminate between the same two nodes, the same key may be used to provide authentication and integrity for both RAS and H.225.0 messages.

Figure 1 shows the most scaleable scenario where both endpoints are within zones that apply the GK-routed model. All the involved GKs share keys mutually. In order to be scaleable, the scenario depicted in Figure 1 is recommended.

NOTE 1 – This scenario does not provide true end-to-end security between endpoints; all security depends on the trusted intermediate gatekeepers.

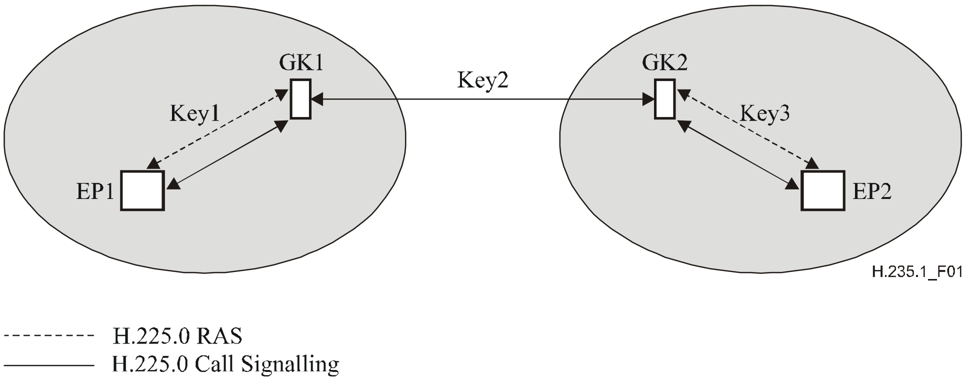

Figure 1/H.235.1 – Illustrating procedure I usage in a GK-GK scenario with both EPs in GK-routed zones

Figure 2 shows a mixed scenario where one EP is within a zone applying the GK-routed model while the other EP is in a zone applying the direct-routed model. This scenario could occur in closed environments where the number of EP2s and GK1s is limited.

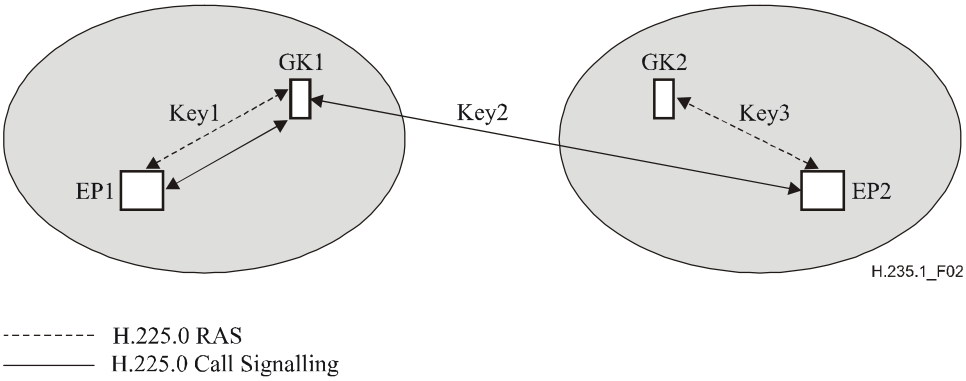

Figure 2/H.235.1 – Illustrating procedure I usage in a mixed scenario with EP1 in a GK-routed zone and EP2 in a direct-routed zone

Figure 3 shows a scenario where both EPs are within zones applying the direct-routed GK model. This scenario is not very scaleable when many EPs are involved. In principle, usage of H.235.2 with procedures II/III is recommended instead. For this specific scenario and procedures I, II or III additional security measures (protecting against call fraud and misuse by means of call authorization with access tokens at H.323 gateways for example), which are not described in this Recommendation, are necessary as well; this is for further study.

NOTE 2 – This scenario provides true end-to-end security among endpoints without relying on trusted intermediate nodes.

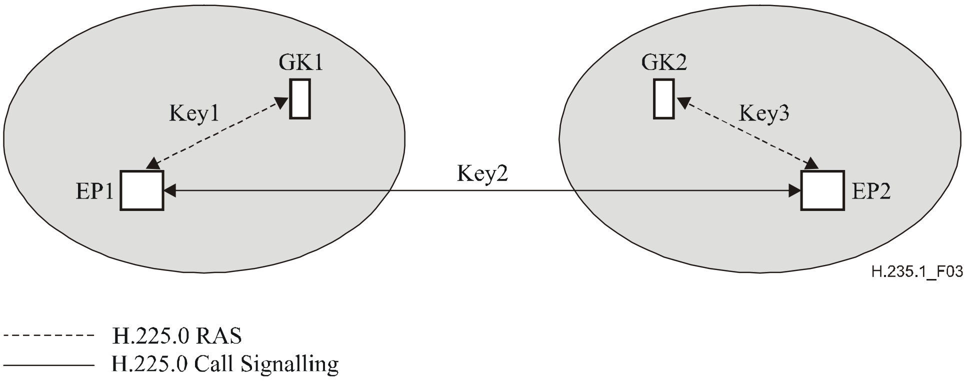

Figure 3/H.235.1 – Illustrating procedure I usage in a scenario with both EPs in zones using a direct-routed GK

Consider the case in Figure 1 where three passwords are pair-wise shared between EP1-GK1, between GK1-GK2 and between GK2-EP2. Three 20-byte keys – *Key1*, *Key2* and *Key3* – are generated from these passwords based on the procedure described in 8.2.4/H.235.0. For maximum security, it is recommended to make each of the three random passwords/keys independent.

Below, we illustrate the procedure details for RAS, H.225.0 and H.245 message authentication and integrity. The description example depicts specific parameters in a GK-routed model; other useful and valid combinations of object identifiers in different scenarios are possible as well.

NOTE 3 – The scenarios shown in Figures 1 to 3 do not scale well in the case where the number of shared symmetric keys (passwords) between GKs (Figure 1), between GKs and remote EPs (Figure 2), or between the EPs (Figure 3) becomes too large.

9.1 RAS message authentication and integrity

Consider the case where EP1 wishes to send an RAS message, say an **ARQ** message, to GK1. EP1 generates a timestamp and a sequence number and includes it in the **timeStamp** and **random** fields respectively, along with GK1's alias in the **generalID** and the EP's ID in the **sendersID** field. These fields are present in the **ClearToken** field of **hashedVals** present in the **cryptoHashedToken** of the **CryptoToken** field of the **cryptoH323Token** of the **ARQ** message.

The **tokenOID** within the **cryptoHashedToken** is set to "A", indicating that all the fields in the **ARQ** message are hashed. The **HASHED** within **token** in **cryptoHashedToken** has **algorithmOID** set to "U" indicating the use of HMAC-SHA1-96 and **params** set to NULL. EP1 then computes the authenticator based on the HMAC-SHA1-96 using the 20-byte key *Key1*. The authenticator is computed over the entire RAS message.

EP1 includes the computed authenticator within **hash** in the **token** field of the **cryptoHashedToken** field of the **CryptoToken** present in the **cryptoH323Token** of the **ARQ** message. The **ARQ** message is then sent to GK1.

Upon receiving the **ARQ** message, GK1 verifies the authenticator based on several criteria that include:

- liveness of the **timestamp**, uniqueness of the **random**;
- identity of the **generalID** and own identifier;
- matching of authenticator in **ARQ** message with that computed by GK1.

9.2 H.225.0 message authentication and integrity

Consider the case where EP1 wishes to send an H.225.0 message, say a **Setup** message, to EP2. EP1 generates a timestamp and a sequence number and includes it in the **timeStamp** and **random** fields respectively, along with GK1's alias in the **generalID** and the EP's ID in the **sendersID** field. EP1 computes also a Diffie-Hellman half-key and includes the Diffie-Hellman parameters **halfkey**, **modsize** and **generator** in the **dhkey** field of the **ClearToken**. These fields are present in the **ClearToken** field of **hashedVals** present in the **cryptoHashedToken** of the **CryptoToken** field of the **cryptoH323Token** of the **Setup** message.

The **tokenOID** within the **cryptoHashedToken** is set to "A", indicating that all the fields in the H.225.0 call signalling message are hashed. The **HASHED** within **token** in **cryptoHashedToken** has **algorithmOID** set to "U" indicating the use of HMAC-SHA1-96 and **params** set to NULL. EP1 then computes the authenticator based on the HMAC-SHA1 algorithm using the 20-byte key *Key1*. The authenticator is computed according to the hash method chosen (A) taking into account the entire H.225.0 call signalling message.

EP1 includes the computed authenticator within **hash** in the **token** field of the **cryptoHashedToken** field of the **CryptoToken** present in the **cryptoH323Token** of the **Setup** message. The **Setup** message is then sent to GK1.

Upon receiving the **Setup** message, GK1 verifies the authenticator based on several criteria that include:

- liveness of the **timestamp**, uniqueness of the **random**;
- identity of the **generalID** and own identifier;
- verification of Diffie-Hellman parameters, e.g., testing whether the 1024-bit prime and generator are correct. Testing of whether the DH-parameters are secure is a time-consuming process and may be done only when local policy requires it;
- matching of authenticator in **Setup** message with that computed by GK1.

If the authenticator is successfully verified, GK1 computes a new authenticator to insert (replace) in the **Setup** message before forwarding it to GK2 as follows. GK1 replaces the **timeStamp**, **random**, **sendersID** and **generalID** fields in the **ClearToken** field of **hashedVals** using values relevant to the GK1-GK2 leg. The **timestamp** field contains the current timestamp, the **random** field contains the next monotonically increasing sequence number for the GK1-GK2 leg, the **generalID** field contains the alias of GK2 and the **sendersID** contains the alias of GK1. GK1 includes also the received Diffie-Hellman parameters into the **dhkey** field of the **ClearToken**.

GK1 then computes a new authenticator for this H.225.0 call signalling message using key *Key2* and algorithm HMAC-SHA1-96 (**algorithmOID**="U"), inserts it in **hash** within **token** and passes the **Setup** message on to GK2.

Upon receiving the **Setup** message, GK2 verifies the authenticator, computes a new authenticator after modifying the **ClearToken** fields in **hashedVals** suitably, inserts it in the **hash** field and passes the **Setup** message on to EP2.

9.3 H.245 message authentication and integrity

Consider the case where EP1 wishes to send an H.245 message, say a **TerminalCapabilitySet** message, to EP2. EP1 checks to see if an H.225.0 message needs to be sent to GK1. If so, then the H.245 message is tunnelled within that H.225.0 message. The fields within the H.225.0 message are set as described earlier for the transmission of an H.225.0 message. Since the H.245 message is tunnelled, the **h323-uu-pdu** in the **h323-UserInformation** message has its fields set as follows:

- **h323-message-body** field is set to the H.225.0 message type that is being transmitted.
- **h245Tunnelling** set to TRUE.
- **h245Control** contains the H.245 PDU octet string.

EP1 generates a **CryptoToken** for the H.225.0 message, sets **tokenOID** to "A", indicating authentication and integrity, sets **timeStamp**, **random**, **sendersID**, **generalID** and **tokenOID** to "T" in the **ClearToken** of the **hashedVals**, set **algorithmOID** to "U", indicating the use of HMAC-SHA1-96 and **hash** to the computed hash authenticator over all the fields of the H.225.0 call signalling message.

However, if no H.225.0 message transmission is pending, then the H.245 message is tunnelled within an ad hoc H.225.0 **facility** message. The **h323-uu-pdu** in the **h323-UserInformation** message has its fields set as follows:

- **h323-message-body** field is set to **facility** which contains:
  - **reason** set to **undefinedReason**;
  - **tokens** and **cryptoTokens** set as for any H.225.0 message.
- **h245Tunnelling** set to TRUE.
- **h245Control** contains the H.245 PDU octet string.

As described above, EP1 generates a **CryptoToken** as part of the H.225.0 **facility** message. The **facility** message is then transmitted by EP1 to GK1.

In either case (whether a H.225.0 message transmission is pending or an ad hoc H.225.0 **facility** message is used), GK1 verifies the authenticator upon receiving the message. Then, if an H.225.0 message transmission is pending for the GK1-GK2 leg, the H.245 message is tunnelled within that message; otherwise, it is tunnelled within an ad hoc H.225.0 **facility** message. As in the case of transmission of any H.225.0 message, a new authenticator is computed for the H.225.0 message prior to its transmission from GK1 to GK2. The process repeats for the GK2-EP2 leg.

9.4 Direct-routed scenario

Secured H.323 entities may communicate not only within the GK-routed environment as outlined in this Recommendation, but may also deploy the direct-routed model. This direct-routed model requires additional security measures (access tokens) that are not necessary in the simpler GK-routed environments. ITU-T Rec. H.235.4 describes how to secure the direct-routed model.

10 Back-end-service support

Secured H.323 entities may use back-end services according to the procedure described in I.1.6/H.235.0.

11 H.235 version 1 compatibility

While these security profiles are developed with ITU-T Rec. H.235 version 2 (ITU-T Rec. H.235 (2000)) in mind, it is also possible to apply the security profiles for ITU-T Rec. H.235 version 1 (ITU-T Rec. H.235 (1998)) with some minor modifications. A recipient is able to detect the presence of the sender's H.235 protocol version by evaluating the security profile object identifiers (see clause 15).

ITU-T Rec. H.235 version 1 (ITU-T Rec. H.235 (1998)) implementations:

- do not set or evaluate the **sendersID** in the **ClearToken**.
- cannot use backend services as in clause 10.

12 Multicast behaviour

H.225.0 multicast messages such as GRQ or LRQ shall not include a CryptoToken according to procedure I. When such messages are sent unicast, then the message shall include a CryptoToken.

13 List of secured signalling messages

This clause provides a summary of how, and by which means, this Recommendation secures the various H.323 signalling messages.

13.1 H.225.0 RAS

H.225.0 RAS message	H.235 signalling fields	Authentication and integrity
Any	cryptoTokens	Procedure I

13.2 H.225.0 call signalling

H.225.0 call signalling message	H.235 signalling fields	Authentication and integrity
Alerting-UUIE, CallProceeding-UUIE, Connect-UUIE, Setup-UUIE, Facility-UUIE, Progress-UUIE, Information-UUIE, ReleaseComplete-UUIE, Status-UUIE, StatusInquiry-UUIE, SetupAcknowledge-UUIE, Notify-UUIE	cryptoTokens	Procedure I

13.3 H.245 call control

H.245 messages to and from secured H.323 entities shall either be piggy-backed as part of the secured fast-connect, or shall be tunnelled using the secured H.225.0 **Facility-UUIE**.

14 Usage of sendersID and generalID

The **ClearToken** holds **sendersID** and **generalID** fields. When identification information is available, the **sendersID** shall be set to the gatekeeper identifier (GKID) for the gatekeeper-initiated message and to the endpoint identifier (EPID) for the endpoint-initiated messages. When identification information is available, the **generalID** shall be set to the GKID for endpoint-initiated messages and to EPID for the gatekeeper-initiated messages. When the identification information is not available, or in case of broadcast/multicast is ambiguous, the field is missing or shall contain a null string. Table 2 summarizes the situation.

Table 2/H.235.1 – Usage of sendersID and generalID

Message	sendersID	generalID
Unicast **GRQ**	**EPID** if available, otherwise **NULL**	**GKID**
Multicast **GRQ**	**EPID** if available, otherwise **NULL**	
GCF, **GRJ**	**GKID**	**EPID** if available, otherwise **NULL**
Initial **RRQ**	**EPID** if available, otherwise **NULL**	**GKID**
RCF	**GKID**	**EPID**
RRJ	**GKID**	
URQ, UCF, URJ, BRQ, BCF, BRJ, DRQ, DCF, DRJ, NSM, RIP, SCI, SCR, XRS (EP-to-GK)	**EPID**	**GKID**
URQ, UCF, URJ, BRQ, BCF, BRJ, DRQ, DCF, DRJ, NSM, RIP, SCI, SCR, XRS (GK-to-EP)	**GKID**	**EPID**
ARQ, IRQ, RAI	**EPID**	**GKID**
ACF, ARJ, BCF, LCF, LRJ, IRR, IRQ, RAC, LCF, LRJ, IACK, INAK	**GKID**	**EPID**
Unicast **LRQ** (EP-to-GK)	**EPID**	**GKID**
Unicast **LRQ** (GK-to-GK)	**GKID**	**GKID**
Multicast **LRQ**	**EPID**	
NOTE – GKID stands for gatekeeper identifier, EPID stands for endpoint identifier. Blank indicates a missing or null identification string.		

15 List of object identifiers

Table 3 lists all the referenced OIDs (see also [OIW] and [WEBOIDs]). There are object identifiers for H.235v1 and for H.235v2.

Table 3/H.235.1 – Object identifiers

Object identifier reference	Object identifier value(s)	Description

"A"	{itu-t (0) recommendation (0) h (8) 235 version (0) 2 1}
	{itu-t (0) recommendation (0) h (8) 235 version (0) 1 1}
		Used in procedure I for the CryptoToken-tokenOID, indicating that the hash includes all fields in the H.225.0 RAS and call signalling message (authentication and integrity).

"E"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 9}
	{itu-t (0) recommendation (0) h (8) 235 version (0) 2 9}
		End-to-end ClearToken carrying sendersID for verification at the recipient side.

"T"	{itu-t (0) recommendation (0) h (8) 235 version (0) 2 5}
	{itu-t (0) recommendation (0) h (8) 235 version (0) 1 5}
		Used in procedures I and IA as the baseline ClearToken for the message authentication and replay protection and optionally also for Diffie-Hellman key management as described in 8.5/H.235.6 clause.

"U"	{itu-t (0) recommendation (0) h (8) 235 version (0) 2 6}
	{itu-t (0) recommendation (0) h (8) 235 version (0) 1 6}
		Used in procedure I for the Algorithm OID, indicating use of HMAC-SHA1-96.

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