INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

J.95

(09/99)

SERIES J: TRANSMISSION OF TELEVISION, SOUND PROGRAMME AND OTHER MULTIMEDIA SIGNALS

Ancillary digital services for television transmission

Copy protection of intellectual property for content delivered on cable television systems

ITU-T Recommendation J.95

(Previously CCITT Recommendation)

ITU-T J-SERIES RECOMMENDATIONS

TRANSMISSION OF TELEVISION, SOUND PROGRAMME AND OTHER MULTIMEDIA SIGNALS

General Recommendations	J.1–J.9
General specifications for analogue sound-programme transmission	J.10–J.19
Performance characteristics of analogue sound-programme circuits	J.20–J.29
Equipment and lines used for analogue sound-programme circuits	J.30–J.39
Digital encoders for analogue sound-programme signals	J.40–J.49
Digital transmission of sound-programme signals	J.50–J.59
Circuits for analogue television transmission	J.60–J.69
Analogue television transmission over metallic lines and interconnection with radio-relay links	J.70–J.79
Digital transmission of television signals	J.80–J.89
Ancillary digital services for television transmission	**J.90–J.99**
Operational requirements and methods for television transmission	J.100–J.109
Interactive systems for digital television distribution	J.110–J.129
Transport of MPEG-2 signals on packetised networks	J.130–J.139
Measurement of the quality of service	J.140–J.149
Digital television distribution through local subscriber networks	J.150–J.159

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

ITU-T RECOMMENDATION J.95

COPY PROTECTION OF INTELLECTUAL PROPERTY FOR CONTENT DELIVERED ON CABLE TELEVISION SYSTEMS

Summary

This Recommendation describes the necessary requirements for a system to protect the intellectual property rights (IPR) of television programming entities against the illegal copying, duplication and distribution of their creative property. The system described herein has aspects that prohibit unauthorized individuals from accessing encrypted MPEG data streams. Also, techniques for "watermarking" television signals for identification and copying allowances are presented.

The material herein contains both general descriptions and discussions of specific technical approaches to copy protection.

Source

ITU-T Recommendation J.95 was prepared by ITU-T Study Group 9 (1997-2000) and was approved under the WTSC Resolution No. 1 procedure on 16 September 1999.

Keywords

Conditional access, digital television, MPEG, security, television, video recording.

FOREWORD

ITU (International Telecommunication Union) is the United Nations Specialized Agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of the ITU. The 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 Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 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.

INTELLECTUAL PROPERTY RIGHTS

The ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. The 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, the 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

Recommendation J.95

COPY PROTECTION OF INTELLECTUAL PROPERTY FOR CONTENT DELIVERED ON CABLE TELEVISION SYSTEMS

(Geneva, 1999)

1 Introduction and background

The illegal recording and duplicating of television intellectual property has resulted in a very large illegal business worldwide, and has cost the owners of the intellectual property significant funds in lost revenue. With the change to digital MPEG television, the problem is exacerbated because digital recordings can be duplicated in their original quality over many generations, whereas analogue recordings are reduced in fidelity with each successive generation, and become unusable at some point. In systems where the MPEG digital signal is received and rendered into an analogue equivalent for viewing on an analogue-only television receiver, the quality of that analogue signal causes it to be a target for pirating also, and thus must be protected.

To assist in these goals, approaches have been developed for hiding digital markings in digital television intellectual property in a manner that is both undetectable and incorruptible. This process, called "watermarking", is based on the science of cryptography but is not cryptographic in itself, and it contains the identity of the property owner and that owner's rules regarding copies, namely, none, one copy for personal use, or unlimited copies.

In addition to watermarking, copy protection requires that clear-text MPEG digital television signals, or their analogue equivalents, are never allowed to traverse signal lines outside of the physical boundaries of the in-home consumer electronics equipment. To accommodate this requirement, a secondary scrambling system is required to temporarily cover these signals during their in-premises distribution, whether in digital or analogue format. A cryptographic system is proposed to cover the digital MPEG signals, and an existing commercial system which rapidly varies the timing is proposed to cover the analogue signals.

To accommodate legal action against any one who would subvert these countermeasures, it is desired that the processes be proprietary and protected by licensing procedures. However, the licensing should be carefully crafted so as not to produce unnecessary discrimination and/or disadvantage to companies who must produce these copy protection systems.

2 Scope

This Recommendation describes both the cryptographic techniques to protect access to clear-text MPEG digital television signals, and a process, known as "watermarking" which indelibly marks the intellectual property as to its owner and that owner's requirements regarding copying. A successful copy protection system supports the legal privilege of the owner of the IPR to control the distribution of the protected product.

Approaches to copy protection for content described in this Recommendation should be considered for use in other applications that require similar protection, such as over-the-air broadcasting, distribution of recorded programmes (e.g. by DVD), etc.

3 References (Informative)

- 1394 Trade Association, Specification for AV/C Digital Interface Command Set.
- Digital Transmission Protection License Agreement, Development and Evaluation License, Digital Transmission Licensing Authority.
- Digital Transmission Licensing Administrator, 5C Digital Transmission Content Protection Specification, Volume 1, Version 0.91.

- Digital Transmission Licensing Administrator, *5C Digital Transmission Content Protection Specification*, Volume 2, Version 0.90.
- IEC 61883-1 (1998), *Consumer audio/video Equipment – Digital Interface – Part 1: General*.
- IEEE Std 1394-1995, *IEEE's Standard for a High Performance Serial Bus*.
- IEEE P1363, *Editorial Contribution to Standard for Public Key Cryptography*, Preliminary Draft, P1363/D3 (May 11, 1998).
- National Institute of Standards and Technology (NIST), *Secure Hash Standard (SHS)*, FIPS Publication 180-1, 17 April, 1995.
- Toshiba Corporation, *Efficient Implementation of an Elliptic Curve Cryptosystem* (available at http://www.dtcp.com).

4 Definitions

This Recommendation defines the following terms:

4.1 **algorithm**: A mathematical process that can be used for the scrambling and descrambling of a data stream.

4.2 **authentication**: The process intended to allow the system to check with certainty the identification of a party.

4.3 **authorization coding**: A digital word that describes the personality or service access capability of the subscriber decoder unit.

NOTE – This code word, which is based on the service access authorized by the billing system, determines which keys are distributed to each customer, and is required at the subscriber decoder to authorize the descrambling of any specific programme.

4.4 **Conditional Access system (CA)**: The complete system for ensuring that cable services are accessible only to those who are entitled to receive them, and that the ordering of such services is not subject to modification or repudiation.

4.5 **cryptanalysis**: The science of recovering the plaintext of a message without access to the key (to the electronic key in electronic cryptographic systems).

4.6 **cryptographic duty cycle**: The maximum secure capacity of a cryptographic process, based on the total number of bits that can be securely encrypted before it becomes advisable to change the key.

4.7 **descrambling**: The process of reversing the scrambling function (see "scrambling") to yield usable pictures, sound, and data services.

4.8 **electronic key**: The term for data signals that are used to control the descrambling process in subscriber decoders.

NOTE – There are at least three types of electronic keys: those used for television signal streams, those used for protecting control system operations, and those used for the distribution of electronic keys on the cable system. See also "authorization coding" which is also effectively a key.

4.9 **encryption**: The process of scrambling signals to avoid unauthorized access.

4.10 **full period terminated service**: A subscription service that is always available to subscribers during the operating hours of the delivery system.

NOTE – By contrast, other services, such as a pay-per-view feature film, are only available for a specific period of time.

4.11 **host**: A device with generalized functionality where modules containing specialized functionality can be connected.

4.12 **integrity**: The ability of a function to withstand being usurped for unauthorized use, or modified to yield unauthorized results.

4.13 **intrusion resistance**: The ability of a hardware object to deny physical, electrical, or irradiation-based access to internal functionality by unauthorized parties.

4.14 **module**: A small device, not working by itself, designed to run specialized tasks in association with a host.

4.15 **non-repudiation**: A process by which the sender of a message (e.g. a request on a pay-per-view) cannot deny having sent the message.

4.16 one-way hash: A mathematical process or algorithm whereby a variable length message is changed into a fixed-length digital word, such that it is very difficult to calculate the original message from the word, and also very difficult to find a second message with the same word.

4.17 pay-per-view: A payment system whereby the subscriber can pay for an individual programme or specified period of time.

4.18 piracy: The act of acquiring unauthorized access to programmes, usually for the purpose of reselling such access for unauthorized reception.

4.19 public key cryptography: A cryptographic technique based upon a two-key algorithm, private and public, wherein a message is encrypted with the public key but can only be decrypted with the private key. Also known as a Private-Public Key (PPK) system.

NOTE – Knowing the public key does not reveal the private key.

Example: Party A would devise such a private and public key, and send the public key openly to all who might wish to communicate with Party A, but retain the private key in secret. Then, while any who have the public key can encrypt a message for Party A, only Party A with the private key can decrypt the messages.

4.20 scrambling: The process of using an encryption function to render television and data signals unusable to unauthorized parties.

4.21 secure signature: A mathematical process by which the origin and integrity of a transmitted message can be ascertained.

NOTE – If a secure signature system is used, the originator cannot deny having sent the message, and the receiver can determine if the message has been modified.

4.22 transport stream: An MPEG-2 Transport Stream.

5 Watermarking of digital television intellectual property

5.1 Background and functional requirements

One of the basic requirements for defining intellectual property is that it must be marked in some manner which identifies it as such and states the identity of the owner of that property. In printed material this is commonly accomplished by the universally recognized trademark symbol, with a footnote indicating the owner of the property. This may often be followed by a statement that delineates the owner’s directions as to its use, for instance "May be copied for non-commercial use only". All currencies use several methods to separate legal tender from counterfeit.

It is the desire of the producers of television property to be able to mark their intellectual property in some manner that delineates their ownership and their imposed limitations on its use. The requirements for the watermarking system are as follows:

1) Clearly state the owner of the property and the copying allowances related thereto.

2) The marking of the television product must be present in all or virtually all frames.

3) The marking must be undetectable in the presentation of the artistic product, even subliminally.

4) The unauthorized changing of the marking must be virtually impossible without corrupting the original product to beneath commercial fidelity.

5) The marking of the product must be machine readable.

6) The false positive error ratio must be negligibly small for long periods (e.g. 1 second in 30 years).

7) The embedded data must be detectable through adaptations that change the screen format or during zoom functions.

8) Multiple watermarks must be able to coexist without interference.

How these functions are provided in a watermarking system are the object of considerable work, two approaches of which are shown in Appendices I and II.

5.2 Implications for design approach

These functional requirements suggest that the marking of the television intellectual property can best be accomplished through using some aspects of the science of cryptology. Using different techniques, the owner of the intellectual property can be highly certain that the marking is difficult to detect, virtually impossible to modify, can easily be inserted into each frame, if required, and can be read by properly authorized origination and consumer electronics hardware. The impact on the original video in using a cryptographic approach is the very slight increase in the noise floor of the signal. The selection of a set of common cryptographic processes and the choice as to how those algorithms are implemented in equipment are not part of this Recommendation.

6 Access control copy protection measures

6.1 Background and functional requirements – Analogue signals

Copy-protected analogue signals that are transported over interconnect wiring at the viewer premises after having been received from a secondary cable system, whether originally sent in analogue format or converted to analogue format from digital in the set-top box, are to be protected from casual copying using one or a combination of more than one currently available schemes to prevent the copying of analogue signals on a conventional VCR. Signals that have no implied copy protection, or those which are converted for use in the display device and are not available in clear-text format outside of the display device, do not need this protection.

6.2 Background and functional requirements – MPEG digital signals

The primary objective of this Recommendation is to assure that clear-text MPEG digital television signals are not easily accessed for the purpose of making unauthorized recordings thereof. This means that all MPEG signals, while being delivered to the customer premises on a cable system, or when being transferred among consumer electronics devices within said premises, must be provided protection against unauthorized access. The science of conditional access, found in Recommendation J.93, applies to those MPEG signals that are being delivered on the secondary cable system. Copy protection applies to those signals that are being transferred among consumer electronic devices within the customer premises.

A system providing encryption and de-encryption of the MPEG signals for transport over the interconnecting wiring in the customer premises is indicated to fulfil this requirement. The encryption system selected must incorporate the following attributes:

1) simple and cost-effective to implement in consumer grade hardware;

2) self-recovering if cryptographic synchronization is lost;

3) implemented in both internal and Point-of-Deployment (POD) module format;

4) can be authorized and de-authorized from, and report status to, a distant point;

5) no purposeful or inadvertent single-point failure can fail the system to clear text MPEG on the interconnecting wiring.

6.3 Authorization centre functionality

With the above-described cryptographic system used to protect signals on the interconnect wiring, an external system element is required to perform the following functions:

1) authorize and provide key for newly installed consumer electronics hardware;

2) de-authorize illegal or stolen consumer electronics hardware;

3) assist in failure recovery modes;

4) provide a security and operational auditing system;

5) provide reports to the owners of affected copy-protected materials;

6) coordinate intended actions with cable system head ends.

Operating efficiencies require these authorization centres to be at least multiple in scope. They may be geographically regional, or encompass smaller populations. They will require a two-way communications link into every customer home and to every device fitted with copy protection functionality. Authorization and failure mode resolution functions will require virtually real-time access to the consumer electronics hardware. Cost and practicality will require this communications to be carried over incumbent networks without unduly burdening existing operations.

In addition, communication paths will be required among the several authorization centres, with back channels to those television providers who pay for the copy protection of their materials. The nature of these back channels are unknown and will require future work to define.

7 Factors regarding the inclusion of copy protection functionality into cable television and consumer electronics equipment

7.1 Watermarking

Since watermarking is installed in the root video at the time of production, and is only read by certain consumer hardware at the point of use, the delivery network, *per se,* has no responsibilities relative to this functionality.

7.2 Access control copy protection

It is in the protection of the signals on the interconnect wiring in the customer premises where the greatest impact to cable delivery systems will occur. Each piece of video processing equipment at the end-user premises must be capable of being authenticated by the authorization facility and with all other germane co-located consumer devices through which MPEG television is to be processed, and it must have the capability to encrypt and decrypt MPEG television signals as required. Any source or recording devices among the consumer hardware, such as DVD players, VCRs, or set-top boxes, must also have the ability to read the watermarking signal so as to determine the owners desire regarding recording. These functions must be standardized as they must work cross-manufacturer and cross-industry as to media.

7.3 Other impacts

The described system requires real time, full-duplex control channels between the regional control centres and all impacted hardware within that region. The medium to be utilized and the protocols required are unknown at this time and the definition of such must result from further work.

Appendix I

The EBU approach to copy protection of television intellectual property delivered on secondary cable TV systems

In order to ensure respect of IPR related to the TV programmes, the proposal made here by the EBU is that it must be possible to identify any digital object and to link that identification to a database which holds all necessary data on the relevant intellectual property rights. By matching this request with the state of the art in digital media protection, the EBU came out with the following remarks and proposes the appended Reference Model (see Figure I.1).

1) No digital object, e.g. TV sequence, should be made available to the public without appropriate protection inserted inside the object itself; ideally in the long term, a non-identified object could not be sold on the digital marketplace.

2) As right holders need different types of information, it appears impossible to satisfy all requests with direct marking of the related data: an identifier should therefore be used only as a link to a secured database containing all the necessary information.

3) Any small part of the sequence which can be isolated and re-used must carry the identifier which is a link to the IPR database.

4) The shortest unique identifier has been established as 64 bits, allowing for a number of combinations from 16 decimal figures to a mix-up of less letters and figures. An example of identification process with 64 bits is given in ISO 10918-4 (identification of still pictures); ISBN, ISSN, ISRC, ISMN, ISWC or ISAN can also be used within the same 64-bit space.

5) The 64-bit identifier should be marked inside the object itself in such a way that it is both invisible and impossible to erase or modify without producing visible effects.

6) The contents of the 64-bit identifier must be delivered by a Registration Authority and may be called a "License Plate" or LP.

7) The identifier may be used to link to a database containing the IPR information (creation watermark) or the distribution information (distribution watermark)

8) Therefore two watermarks may be inserted inside the data stream, with reference to remarks 5) to 8). In order to mark all frames, if MPEG-2 is used, each I-frame will be marked.

9) As an identifier cannot be fully significant with its small size, it can only be used as a link. The link type can be a hyper link between the object and the database containing the relevant information.

10) To secure the watermark, it has been suggested to duplicate the contents (64 bits) in a tag present in the bit stream file. A 64-bit space has been reserved for this purpose in MPEG-2.

11) It should also be recommended to use the 64-bit space parted in two halves, the first 32 bits defining the originating Registration Authority (REGAUT) who delivered the identifier, while the last 32 bits would be used to number the object, with a capacity of 4 billion identifiers.

12) A currently used such identifier is the IMLP (ISO Multimedia License Plate for still picture identification) which is structured as follows:

/ISO country code (16 bits)/REGAUT identity (16 bits)/registration number (32 bits)/

13) Tables to get the REGAUT URL from their identity will be available on a web site to allow for automatic link between the object and its IPR data.

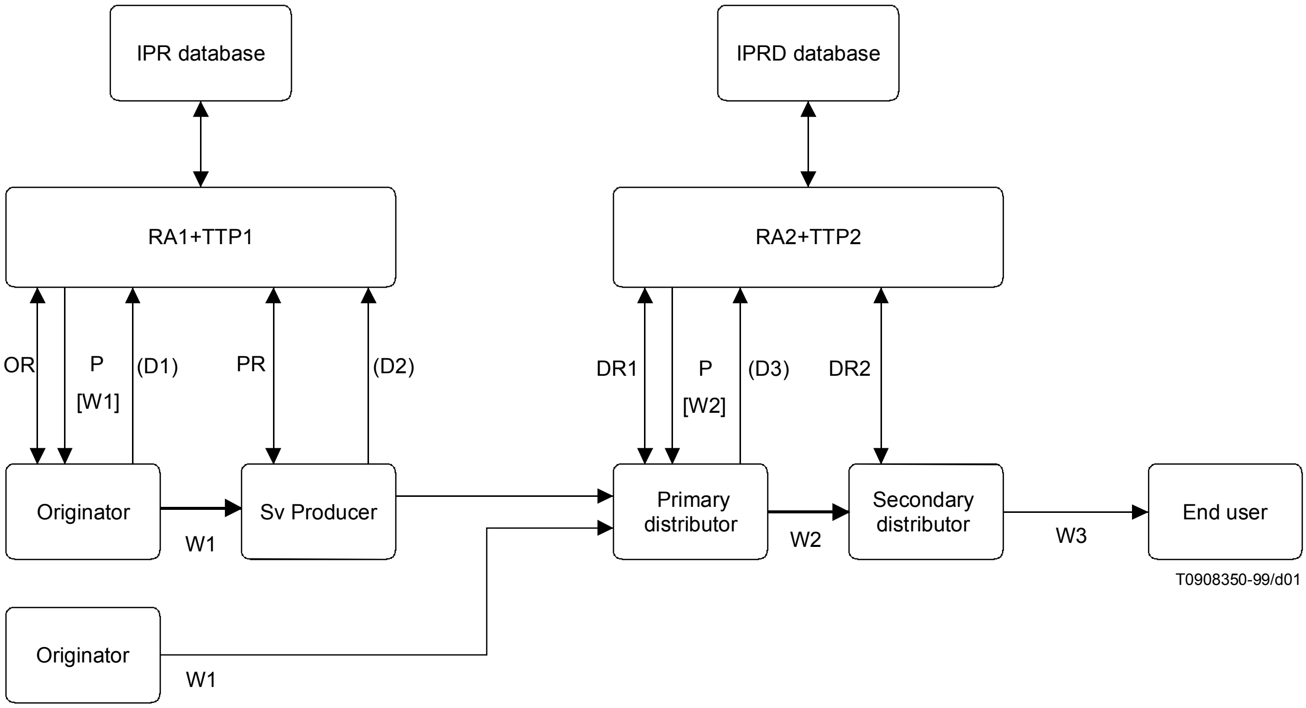
14) Monitoring the bit stream will result in reading the two watermarks, with the ability to link automatically to the databases where the required information is kept.

15) Watermark content can be used for legal purposes, in particular if it is delivered by a Registration Authority.

16) Delivery of the watermark content is done against supply of information which is supposedly reliable and definitely kept in a safe place by the REGAUT.

17) Each REGAUT may define its registration process and must guarantee authenticity of registered data.

18) Access control to the relevant data is up to the REGAUT management.



OR	Originator Registration
P	Protection (Watermark content)
D1	Declaration (of the work related to W1)
PR	Sv Producer Registration
D2	Declaration of the work done (Mise en Forme or Assembly)
DR1	Distributor Registration (Primary)
D3	Declaration (of what was distributed)
DR2	Distributor Registration (Secondary)

Figure I.1/J.95 – IPR Reference Model

Appendix II

Galaxy watermark proposal

Summary

The proposed Galaxy watermarking technology embeds 8 bits of data as a transparent digital watermark into uncompressed digital video. This watermark can be detected in both baseband and MPEG-2 domains. It is called the Primary Mark. The first two bits of Primary Mark represent Copy Control Information (CCI) such as "copy-never", "Copy-one-generation" and "No-more-copies". The detector uses the adaptive period detection algorithm to detect the Primary Mark with a predetermined false positive error ratio. Even from the heavily degraded content, reliable detection can be achieved without exceeding the predetermined false positive ratio, which is set to less than 10^{-12}, in trade with the extension of the detection time.

Galaxy proposes to have another transparent watermark inserted into the video at the digital recorders to serve as an identification of the copied material. This is called the Copy Mark insertion. The technology inserts another transparent watermark without any disturbance to the pre-embedded Primary Mark. The insertion of this Copy Mark can take place in both baseband and MPEG-2 domains and the detection of the same Copy Mark can also be done in both domains. We add Copy Mark to the "Copy-one-generation" content to change the status to "No-more-copies" for the purpose of Generation Copy Control. Note that the Copy Marks embedded in the both domains are identical.

Galaxy watermarking technology is compatible and interchangeable between the MPEG domain and baseband domain; therefore, it offers great deal of freedom to the device manufacturers to choose the implementation location of the watermark detector and the Copy Mark inserter. The final decision of the location to place the detector/inserter should be made from the aspect of security and implementation cost.

Galaxy completed the design and software prototyping of a single unified watermarking technology and has performed extensive survivability and transparency tests to make the algorithm stable. The survivability tests that have been achieved are reported below. The technology is mature enough for an immediate verification test by CPAC. This appendix also describes how the false positive error ratio can be controlled by the threshold value at the detection.

Finally, automated embedding technology and easy-to-operate and secure embedding system is described. The real-time embedding system is a PC-based DSP system.

II.1 System architecture

II.1.1 Overview of watermark usage

In the Galaxy watermarking system, Primary Mark carries 8 bits of information and the first two bits are used for Copy Control Information (CCI). The usage of other bits is beyond the scope of this appendix but should be agreed by the involved parties of ICPAC, including APS trigger bits. The Galaxy technology also can insert and detect another independent watermark called Copy Mark, which coexists with Primary watermark and used to changes the interpretation of CCI of Primary Mark for the Generational Copy Control.

In the usage of CCI, the following is commonly understood at DHSG.

1) Content distributed by electronic means such as digital TV may be marked (1,1), (1,0), (0,0) or may not be marked.

2) All content distributed by DVD-ROM media are either marked (1,1), (0,0) or may not be marked.

3) All content distributed by DVD-ROM media marked (1,1) is scrambled by CSS.

4) DVD playback devices are able to distinguish recordable media from read-only media.

5) "No more copies" state is allowed only to be on recordable media.

Examples of record control and playback control are described below. Galaxy watermark technology is capable and flexible to address all possible implementation scenarios. The actual implementation should be discussed in conjunction with the total copy protection system design.

Record and generation copy control

The detected CCI listed in Table II.1 can be used to trigger action of digital recorders such as DVD recorders. Here CFP is the Call for Proposal issued by DHSG in May of 1997.

Table II.1/J.95 – Definition of CCI and required Response for Copy Control in recording devices

Detected CCI	Definition in CFP	Response of Recorder
1,1	Copy-never	Prevent Copy
1,0	Copy-one-generation	Allow Copy and add Copy Mark
1,0 with Copy Mark	No-more-copies	Prevent Copy
0,0 or no mark	Copy allowed	Allow Copy

Playback control

The detected CCI and the information of the playback media can be used to trigger the actions of compliant DVD players. An example of definition is listed in Table II.2. By assuming that the watermark will not be detected when CSS scrambling is present, we assigned "Prevent playback" as unauthorized copying when CCI = (1,1) is detected from DVD-ROM media without CSS scrambling.

Table II.2/J.95 – Definition of CCI and response for Playback Control in DVD players

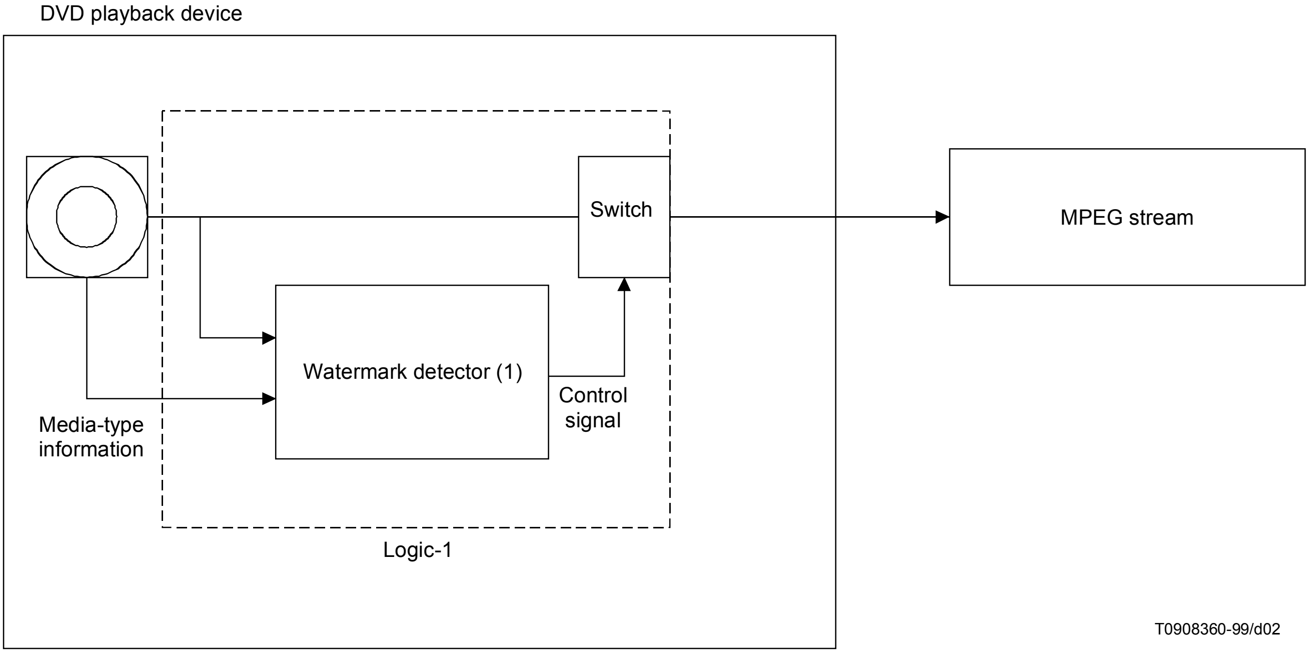
Detected media type	Detected CCI	Response of the device
Read only	1,1	Prevent playback*
	1,0	Prevent playback
	1,0 with Copy Mark	Prevent playback
	0,0 or no mark	Allow playback
Recordable & rewritable	1,1	Prevent playback
	1,0	Prevent playback
	1,0 with Copy Mark	Allow playback
	0,0 or no mark	Allow playback

II.1.2 System Configuration

System configuration for playback control and recording control for DVD system are explained here using three types of watermark detection/Copy Mark insertion logic in this subclause. These are the fundamental building blocks for the total Copy Control system design. The Primary Mark can be detected in both MPEG-2 and baseband domains and the Copy Mark can be inserted and detected in both domains. This feature offers a great deal of freedom to the device manufacturers to choose the implementation location of the watermark detector and the Copy Mark inserter.

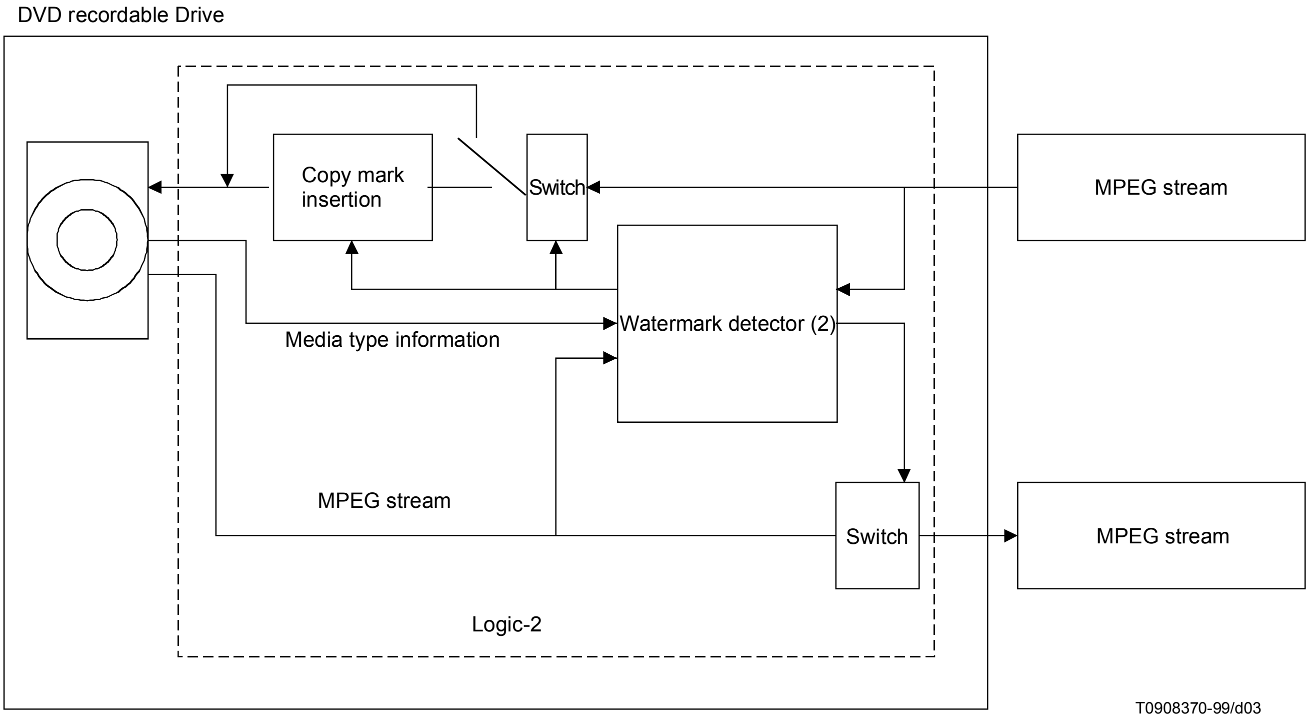
Logic-1: DVD playback control

This logic detects the Primary Mark and Copy Mark directly from the MPEG data. The resulted CCI and Copy Mark will be used with media-type information by the controller of the DVD playback device for playback control, according to the action defined in II.1.1. The box enclosed by the dotted line represents the block diagram of the logic.



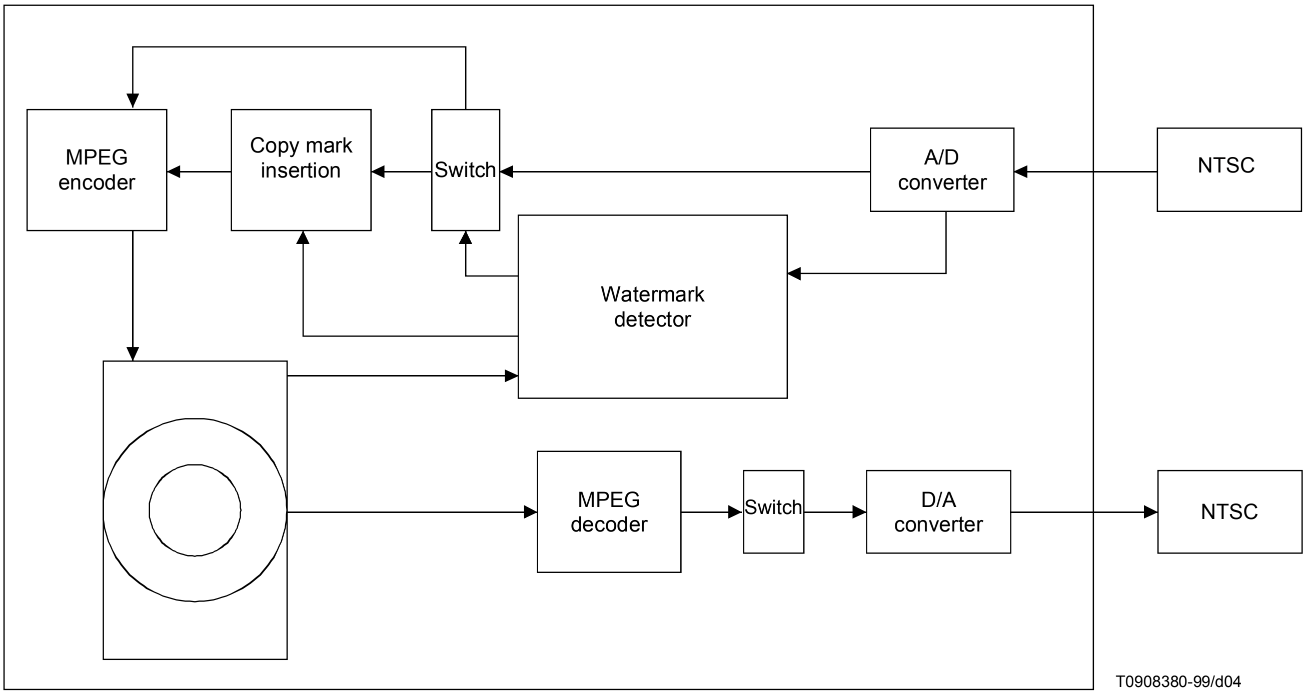
Logic-2: DVD playback, copy and generation copy control

In addition to the playback control, this logic performs the record and generational copy control by using Copy Mark insertion function and the same watermark detector of Logic-1. The media type information is also required for playback. In this logic, all functions are performed directly to MPEG video stream; thus, it can be placed into the DVD drive unit if necessary. The box enclosed by the dotted line represents the block diagram of the logic-2.



Logic-3: Copy control and generation copy control on baseband

This logic is for record control and generational control on recording devices with baseband video input or DVD-based video recorders. The functionality is as same as the input portion of the Logic-2; the difference is that the target video stream is the uncompressed video data. The following figure shows an example of implementation with DVD playback control.



II.2 Generation Copy Control

The Galaxy technology can insert another transparent watermark called Copy Mark without any destruction of Primary Mark in both baseband and MPEG-2 domains and detects it also in both domains. The insertion of Copy Mark at MPEG-2 domain is designed to strictly preserve the packet size of MPEG-2 stream in order to meet the hardware implementation constraint. The presence of the Copy Mark will change the interpretation of CCI (1,0) from "Copy-one-generation" to "No-more-copies" for the purpose of Generation Copy Control as given in Tables II.1 and II.2.

By using the Copy Mark approach, the system can fully support both of the digital and analogue transmission from the installed base set-top box, without requesting any cooperative action to the devices existing on the transmission paths.

II.3 Technical maturity

The member companies of Galaxy have been independently and jointly developing watermarking technology for digital video since early 1996, aiming for the application of the Copy Protection of DVD content. They had already implemented the detection logic by FPGA (programmable logic array) and embedding algorithm by DSP and demonstrated real-time embedding and detection at the DHSG in 1997 and 1998. High survivability of watermark has been shown and remarking technology for generational copy control has been demonstrated at a DHSG live test in February 1998. Galaxy acknowledged each member’s technical expertise and announced the merge of proposals in February 1999.

The Galaxy watermarking technology is mature enough for an immediate verification test by the CPAC. All functions in the Table II.3 have been tested with the transparency, reliability and survivability of Primary Mark (PM) and Copy Mark (CM), which cover both MPEG and baseband domains. Prototype of real-time embedding system will be available soon for studio trial. However, a detailed product schedule is dependent upon the CPAC’s final selection schedule.

Table II.3/J.95 – Availability of required function as of March 1999

		Functional description	**Availability**
WM embedding in baseband video	Studio system	Automated watermark embedding system with full 8-bit information per field Pseudo real-time embedding with ITU-R-656 I/O	Yes
PM detection in MPEG domain	Logic-1, 2	Direct detection from MPEG stream in real time	Yes
PM detection in baseband domain	Logic-3	Detection after A/D conversion	Yes
CM insertion in MPEG domain	Logic-2	CM insertion directly into MPEG stream with preserving MPEG packet size	Yes
CM insertion in baseband domain	Logic-3	CM insertion to the video after A/D conversion	Yes
CM detection in MPEG domain	Logic-1, 2	Direct detection from MPEG stream in real time	Yes
CM detection in baseband domain	Logic-3	Detection after A/D conversion	Yes

II.4 Gate Count analysis

The estimated gate sizes of Logic-1, Logic-2 and Logic-3 in II.2 are listed in Table II.4.

Table II.4/J.95 – Function and gate count summary of Galaxy watermarking detection chips

Logic type	Purpose	Functional description	Gate counts	Target devices
1	Playback control	Primary Mark and Copy Mark watermark detection from MPEG stream	30 k gates 5 kbyte RAM	DVD playback device
2	Playback control Record control Generation Copy Control	Primary Mark and Copy Mark watermark detection and Copy Mark insertion in MPEG stream	35 k gates 5 kbyte RAM	DVD recordable drive
3	Playback control Record control Generation Copy Control	Primary Mark and Copy Mark watermark detection and Copy Mark insertion in baseband domain after A/D conversion	30 k gates 42 kbyte RAM	DVD recordable devices with analogue video input

The estimated gate size can vary depending on the architecture and the resource availability in the semiconductor systems of the recording and/or playback device. The gate size of Logic-3 does not include an analogue-to-digital conversion prior to the detection process.

These gate sizes do not necessarily represent the future product specification, and are subject to change because the detail of function specification may change according to new CPAC requirements.

II.5 Robustness tests

The survivability test was conducted under the conditions:

- Data Payload: 8 bits (arbitrary 256 states can be represented).
- False Positive Error Ratio less than 10^{-12} in 10-second detection period.

These conditions must be specified before any comparison of different technologies because there is a tradeoff relation among transparency, data payload, false positive error ratio and the survivability of the watermark.

Eight bits of data were detected from the test video clips by Galaxy's adaptive detection period algorithm with a maximum of 20 seconds detection window in both MPEG-2 and baseband domains. The 20 sample clips provided by DHSG in 1997 were used and the successive processes of studio video processing → MPEG-2 compression → VHS recording → MPEG-2 recompression were applied in order to simulate the expected degradation in real world. The studio video processing and parameters of each process are listed in Table II.5. Eight bits were detected correctly typically in 1 second or less after the first MPEG-2 compression and within 10 seconds in most of cases even after MPEG-2 recompression.

Table II.5/J.95 – List of Survivability test items

Studio video processing (DVNR-1000)	Note
Brick Wall filter	
Aperture enhancement	
Noise reduction	
98% speed reduction	Drop one frame in each 50 frames
Watermark blending 50%	(reference test)
Letterbox conversion	
Offset letterbox conversion	
Random spatial shifting	More than 10 shifts per 20 s
Hue shifting	30-degree hue shifting

MPEG compression	4 Mbit/s-10 Mbit/s, different GOP, field/frame
VHS recording	3 DNR, TBC on/off, etc.
MPEG recompression	CBR real-time encoder, change of GOP interval

In addition to the simulation test, the following three, real-environment survivability tests were performed:

1) MPEG compression → Satellite transmission → Analogue cable transmission → VHS recording.

2) MPEG compression → Satellite transmission → DirecTV transmission → VHS recording.

3) HD format embedding → Down conversion to SD → Analogue conversion → MPEG compresFalse Positive Analysis.

II.6 False Positive analysis

False positive error occurs when the detector misinterprets a non-marked video segment to be a marked one. The false positive error ratio must be extremely low, e.g. 10^{-12}, since it prevents the device from copying a legitimate copy. The Galaxy technology can control the expected false positive error ration by a predetermined watermark detection threshold.

Brief description of algorithm

Galaxy's proposed algorithm detects 8 bits of CCI in each field from MPEG-2 stream or baseband video in the following way. First, the detected watermark strength of each bit is calculated by summing up outputs observed from sub-blocks assigned in the field. Then all the bits are interpreted when the strength of every bit exceed a predetermined threshold value. Here, let ε_B be a probability that the detection strength of a single bit would exceed the threshold. Given that the detection values from all 8 bits are independent from each other, the probability of all 8 bits to exceed the threshold in an unmarked frame is represented as follows:

$$\varepsilon = \varepsilon_B{}^8$$

According to the Central Limit Theorem, the distribution of the signal strength observed in the unmarked frames (i.e. noise strength) can be treated as a normal distribution and its variance can be calculated based on each variance of the outputs. This is because the strength is a linear summation of large number of the random outputs. Therefore, the probability that the normalized noise strength R would exceed the threshold value T can be estimated by using a normal probability density function as follows:

$$\varepsilon_B = \mathrm{P}(|R| > T) = 2\int_T^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt$$

where "P(x)" indicates a probability of an event "x".

If the normalized signal strength is weaker than the threshold, the signal will be continuously accumulated with the following frames (fields). The signal accumulation will continue on until the accumulated signal strength reaches the threshold (watermark detected), or the accumulation time exceeds the maximum cut-off time (watermark not detected), whichever comes first (Frame Accumulation Detection). Furthermore, because the algorithm chooses appropriate signals so that they are independent from each other, the variance of the accumulated signal can be calculated as a function of the square root of the number of accumulations *f*, and thus the behaviour of the accumulated signal:

$$S_f = \sum_{1}^{f} R_i$$

can be estimated by using a normal probability density function as follows:

$$\varepsilon_B = \mathrm{P}(|\frac{S_f}{\sqrt{f}}| > T) = 2\int_{T}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt$$

Since the algorithm performs independent detection test at the same time of the accumulation, the total false positive error ratio for each bit can be expressed as follows:

$$\varepsilon' = 1 - (1 - \varepsilon)^f$$

where ε is the false positive error ratio for a single test and *f* is the number of accumulations. Table II.6 shows some threshold values *T* and the corresponding false positive errors ratio ε′, where the maximum of accumulations is set to 40.

Table II.6/J.95 – List of false positive ratio vs. threshold

Target ratio (ε′) of false positive errors	**Threshold (*T*) for 8-bit detection**
10^{-8}	1.85878
10^{-9}	1.98372
10^{-10}	2.10306
10^{-11}	2.21752
10^{-12}	2.32729
10^{-13}	2.44118

II.7 Embedding technology and system

In the Galaxy technology, the embedding is a two-path process:

1) image content analysis;

2) luminance modification process.

This two-path process allows automated control of embedding strength to satisfy required transparency, robustness and false positive error ratio. Eight bits of CCI data are embedded as Primary Mark into each field of uncompressed digital video.

The studio embedding system is a PC-based DSP system that is able to run in real time with frames of delay. Its input and output is ITU-R-656 digital video interface. The system is planned to have a real-time monitor to confirm the signal strength of embedded clips and offer easy-to-operate user interface for adjustable embedding parameters. The system also offers security functions such as access control to unauthorized operators and prevention of accidental and intentional re-embedding.

II.8 Abbréviations

A/D	Analogue to Digital
APS	Analogue Protection system
CCI	Copy control Information
CFP	Call for Proposal
CM	Copy Mark
CPAC	Copy Protection Advisory Committee
CSS	Contents Scramble System
D/A	Digital to Analogue
DHSG	Data-Hiding Sub-Group
DSP	Digital Signal Processor
DVD ROM	Digital Versatile Disc Read-Only Memory
DVNR	Digital Video Noise Reduction
FPGA	Field Programmable Gate Array
ICPAC	Interim CPAC
MPEG-2	Moving Pictures Expert Group 2
PM	Primary Mark
WM	Water Mark

II.9 Contact Information

Contact in Japan

IBM Corporation
Tokyo Research Laboratory
1623-14, Shimotsuruma, Yamato-shi
Kanagawa-ken, 242-8502
Japan

NEC Corporation
1-10, Nisshincho, Fuchu-shi
Tokyo, 183-8501
Japan

Hitachi Ltd.
292, Yoshidacho, Totsuka-ku
Yokohama-shi
Kanagawa-ken, 244-0817
Japan

Pioneer Electronic Corporation
1-1, Fujimi 6 chome, Tsurugashima-shi
Saitama-ken, 350-2288
Japan

Sony Corporation
6-7-35, Kitashinagawa, Shinagawa-ku
Tokyo, 141-0001
Japan

Contact in the United States

Director of Licensing Development
IBM Corporation
500 Columbus Avenue
Thornwood, NY 10594
United States of America

NEC Research Institute Inc.
4 Independence Way
Princeton, NJ
United States of America

Appendix III

The 5C proposal for the copy protection of MPEG video intellectual property

Intellectual property

Implementation of this specification requires a license from the Digital Transmission Licensing Administrator.

Contact information

Feedback on this specification should be addressed to spec-comments@dtcp.com .

The Digital Transmission Licensing Administrator can be contacted at dtla@intel.com .

The URL for the Digital Transmission Licensing Administrator web site is: http://www.dtcp.com .

NOTE – The source documents for the material found in this appendix can be acquired from the copyright holders only through execution of a non-disclosure agreement (NDA). Contact the Digital Transmission Licensing Administrator for the proper sources of this information.

III.1 Introduction

III.1.1 Purpose and scope

The *5C Digital Transmission Content Protection Specification* defines a cryptographic protocol for protecting audio/video entertainment content from unauthorized copying, intercepting, and tampering as it traverses digital transmission mechanisms such as a high-performance serial bus that conforms to the IEEE 1394-1995 standard. Only legitimate entertainment content delivered to a source device via another approved copy protection system (such as the DVD Content Scrambling System) will be protected by this copy protection system.

The use of this specification and access to the intellectual property and cryptographic materials required to implement it will be the subject of a license. The Digital Transmission Licensing Administrator (DTLA) is responsible for establishing and administering the content protection system described in this specification.

While DTCP has been designed for use by devices attached to serial buses as defined by the IEEE 1394-1995 standard, the developers anticipate that it will be appropriate for use with future extensions to this standard, other transmission systems, and other types of content as authorized by the DTLA.

III.1.2 Overview

This specification addresses four layers of copy protection:

- **Copy control information (CCI)**

  Content owners need a way to specify how their content can be used ("Copy-one-generation," "Copy-never," etc.). This content protection system is capable of securely communicating copy control information (CCI) between devices in two ways:

  – The encryption mode indicator (EMI) provides easily accessible yet secure transmission of CCI via the most significant two bits of the **sy** field of the isochronous packet header.

  – CCI is embedded in the content stream (e.g. MPEG). This form of CCI is processed only by devices which recognize the specific content format.

- **Device authentication and key exchange (AKE)**

  Before sharing valuable information, a connected device must first verify that another connected device is authentic. To balance the protection requirements of the content industries with the real-world requirements of PC and consumer electronics (CE) device users, this specification includes two authentication levels, Full and Restricted:

  – Full Authentication can be used with all content protected by the system.

  – Restricted Authentication enables the protection of "Copy-one-generation" and "No-more-copies" content only. Copying devices such as digital VCRs employ this kind of authentication.

- **Content encryption**

  Devices include a channel cipher subsystem that encrypts and decrypts copyrighted content. To ensure interoperability, all devices must support the specific cipher specified as the baseline cipher. The subsystem can also support additional ciphers, whose use is negotiated during authentication.

- **System renewability [MP1]**

  Devices that support Full Authentication can receive and process system renewability messages (SRMs) created by the DTLA and distributed with content and new devices. System renewability ensures long-term integrity of the system through the revocation of compromised devices.

Figure III.1 gives an overview of content protection. In this overview, the source device has been instructed to transmit a copy protection stream of content. In this and subsequent diagrams, a source device is one that can send a stream of content. A sink device is one that can receive a stream of content. Multifunction devices such as PCs and record/playback devices such as digital VCRs can be both source and sink devices.

1) The source device initiates the transmission of a stream of encrypted content marked with the appropriate copy protection status (e.g. "Copy-one-generation," "Copy-never," or "No-more-copies") via the EMI bits.[1]

2) Upon receiving the content stream, the sink device inspects the EMI bits to determine the copy protection status of the content. If the content is marked "Copy-never," the sink device requests that the source device initiate Full AKE. If the content is marked "Copy-one-generation" or "No-more-copies" the sink device will request Full AKE, if supported, or Restricted AKE. If the sink device has already performed the appropriate authentication, it can immediately proceed to Step 4.

[1] If content requested by a sink device is protected, the source device may choose to transmit an empty content stream until at least one device has completed the appropriate authentication procedure required to access the content stream.

3) When the source device receives the authentication request, it proceeds with the type of authentication requested by the sink device, unless Full AKE is requested but the source device can only support Restricted AKE, in which case Restricted AKE is performed.

4) Once the devices have completed the required AKE procedure, a content channel encryption key can be exchanged between them. This key is used to encrypt the content at the source device and decrypt the content at the sink.

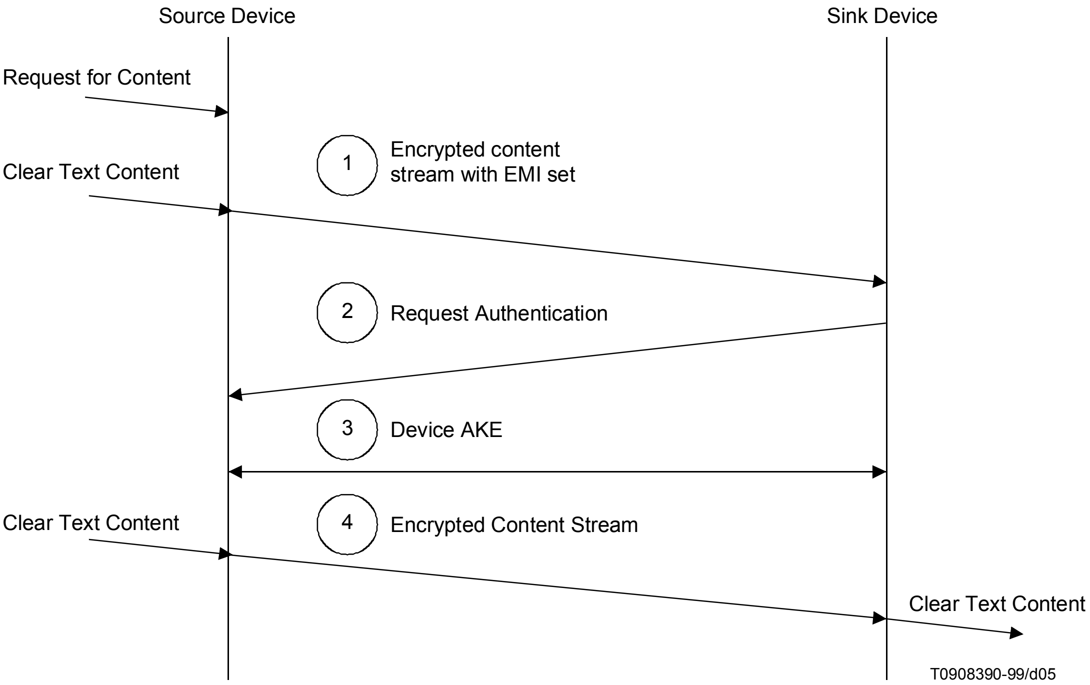


Figure III.1/J.95 – Content Protection overview

III.1.3 References

This specification shall be used in conjunction with the following publications. When the publications are superceded by an approved revision, the revision shall apply.

– Digital Transmission Protection License Agreement, *Development and Evaluation License*, Digital Transmission Licensing Authority.

– 1394 Trade Association, *Specification for AV/C Digital Interface Command Set*.

– Digital Transmission Licensing Administrator, *5C Digital Transmission Content Protection Specification*, Volume 1, Version 0.91.

– Digital Transmission Licensing Administrator, *5C Digital Transmission Content Protection Specification*, Volume 2, Version 0.90.

– Digital Transmission Protection License Agreement, *Development and Evaluation License*, Digital Transmission Licensing Authority.

– IEEE Std 1394-1995, *IEEE's Standard for a High Performance Serial Bus*.

– IEEE P1363, *Editorial Contribution to Standard for Public Key Cryptography*, Preliminary Draft, P1363/D3 (May 11, 1998).

– IEC 61883-1 (1998), *Consumer Audio/Video Equipment – Digital Interface – Part 1: General.*

– National Institute of Standards and Technology (NIST), *Secure Hash Standard (SHS)*, FIPS Publication 180-1, 17 April, 1995.

– Toshiba Corporation, *Efficient Implementation of an Elliptic Curve Cryptosystem,* available from http://www.dtcp.com/

III.1.4 Organization of this appendix

This appendix is organized as follows:

- Subclause III.1 provides an overview of content protection.
- Subclause III.2 lists terms and abbreviations used throughout this appendix.
- Subclause III.3 describes the operation of the overall Digital Transmission Content Protection System as a state machine.
- Subclause III.4 addresses the particulars of the Full Authentication level of device authentication and key exchange.
- Subclause III.5 addresses the particulars of the Restricted Authentication level of device authentication and key exchange.
- Subclause III.6 describes the details of content channel establishment after Full or Restricted Authentication takes place.
- Subclause III.7 describes the System Renewability capabilities.
- Subclause III.8 covers AV/C command extensions.

III.1.5 State machine notation

State machines are employed throughout this apendix to show various states of operation. These state machines use the style shown in Figure III.2.

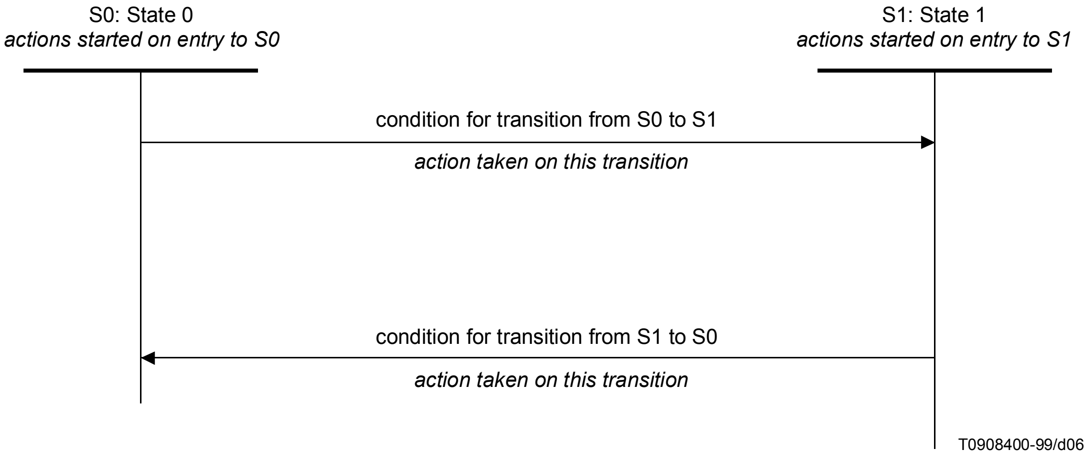

S0: State 0
actions started on entry to S0
S1: State 1
actions started on entry to S1
condition for transition from S0 to S1
action taken on this transition
condition for transition from S1 to S0
action taken on this transition
T0908400-99/d06

Figure III.2/J.95 – State Machine example

State machines make three assumptions:

1) Time elapses only within discrete states.

2) State transitions are instantaneous, so the only actions taken during a transition are setting flags and variables and sending signals.

3) Every time a state is entered, the actions of that state are started. A transaction that points back to the same state will restart the actions from the beginning.

III.1.6 Notation

The following notation will be used:

$[X]_{msb\_z}$ = The most significant z bits of X.

$[X]_{lsb\_z}$ = The least significant z bits of X.

$S_{X^{-1}}[M]$ = Sign M using EC-DSA with private key X^{-1} (Details of signature algorithm are in III.A.4).

V_{X^1} [M] = Verify signature of M using EC-DSA with public key X^1 (Details of the verification algorithm are in III.4).

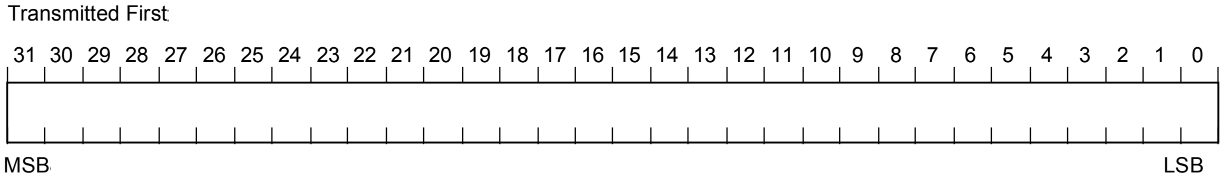
$X \,||\, Y$ = Ordered Concatenation of X with Y.

$X \oplus Y$ = Bit-wise Exclusive-OR (XOR) of two strings X and Y.

III.1.7 Numerical values

Three different representations of number are used in this specification. Decimal numbers are represented without any special notation. Binary number are represented as a string of binary (0, 1) digits followed by a subscript 2 (e.g. 1010_2). Hexadecimal numbers are represented as a string of hexadecimal (0..9, A..F) digits followed by a subscript 16 (e.g. $3C2_{16}$).

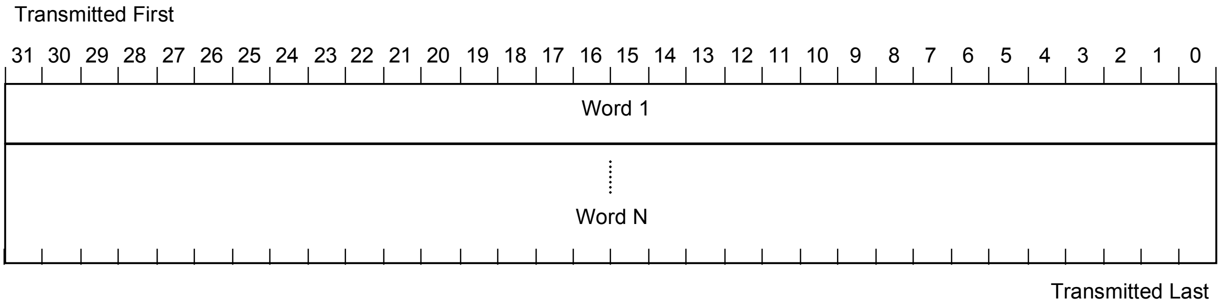
III.1.8 Bit ordering



T0909260-00/d07

Figure III.3/J.95 – Bit ordering

III.1.9 Packet Format



T0909270-00/d08

Figure III.4/J.95 – Packet format

III.1.10 Treatment of optional portions of this Specification

Features of this specification that are labelled as "optional" describe capabilities whose usage has not yet been established by the DTLA.

III.2 Terms and abbreviations

For further study

III.3 The 5C Digital Transmission Content Protection System

III.3.1 Content Source Device

Figure III.5 shows the various states of operation for a device that is a source of content.

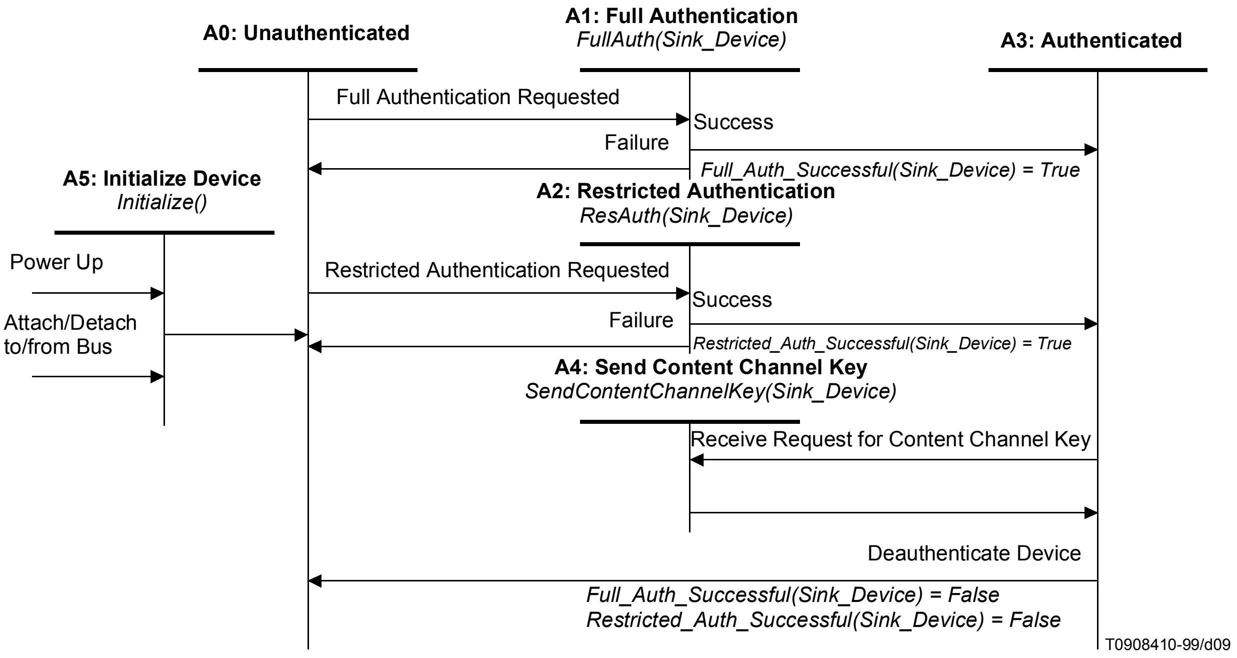

A0: Unauthenticated
A1: Full Authentication
FullAuth(Sink_Device)
A3: Authenticated
Full Authentication Requested
Success
Failure
Full_Auth_Successful(Sink_Device) = True
A5: Initialize Device
Initialize()
A2: Restricted Authentication
ResAuth(Sink_Device)
Power Up
Restricted Authentication Requested
Success
Attach/Detach to/from Bus
Failure
Restricted_Auth_Successful(Sink_Device) = True
A4: Send Content Channel Key
SendContentChannelKey(Sink_Device)
Receive Request for Content Channel Key
Deauthenticate Device
Full_Auth_Successful(Sink_Device) = False
Restricted_Auth_Successful(Sink_Device) = False
T0908410-99/d09

Figure III.5/J.95 – Content Source Device State Machine

A Power up or Attach/Detach to/from the bus event resets this state machine into **State A5: Initialize Device.**

State A5: Initialize Device. In this state, the device is initialized.

Transition A5:A0. This transition to **State A0: Unauthenticated** occurs following the completion of the initialization process.

State A0: Unauthenticated. A device is in an unauthenticated state, waiting to receive a request to perform the Full or Restricted Authentication procedure.

Transition A0:A1. This transition occurs when the device receives a request to perform the Full Authentication procedure with a sink device *(Sink_Device).*

State A1: Full Authentication. In this state, the process *FullAuth(Sink_Device)* is performed. This process is described in detail in III.4.

Transition A1:A3. This transition occurs when *FullAuth(Sink_Device)* has been successfully completed.

Set Full_Auth_Successful(Sink_Device) = True

Transition A1:A0. This transition occurs when *FullAuth(Sink_Device)* is unsuccessful.

Transition A0:A2. This transition occurs when the device receives a request to perform the Restricted Authentication procedure with a sink device *(Sink_Device).*

State A2: Restricted Authentication. In this state, the device executes the process *ResAuth(Sink_Device).* This procedure is described in detail in III.5.

Transition A2:A3. This transition occurs when *ResAuth(Sink_Device)* has been successfully completed.

Set Restricted_Auth_Successful(Sink_Device) = True

Transition A2:A0. This transition occurs when *ResAuth(Sink_Device)* is unsuccessful.

State A3: Authenticated. When a device is in this state, it has successfully completed either the Full or Restricted Authentication procedure.

Transition A3:A4. An authenticated device is requested to send the values necessary to construct a Content Key to a sink device.

State A4: Send Content Channel Key. In this state, the source device sends values necessary to create a content key to an authenticated sink device by executing S*endContentChannelKey(Sink_Device)*. This process is described in III.6.

Transition A4:A3. This transition occurs on completion of the process *SendContentChannelKey(Sink_Device).*

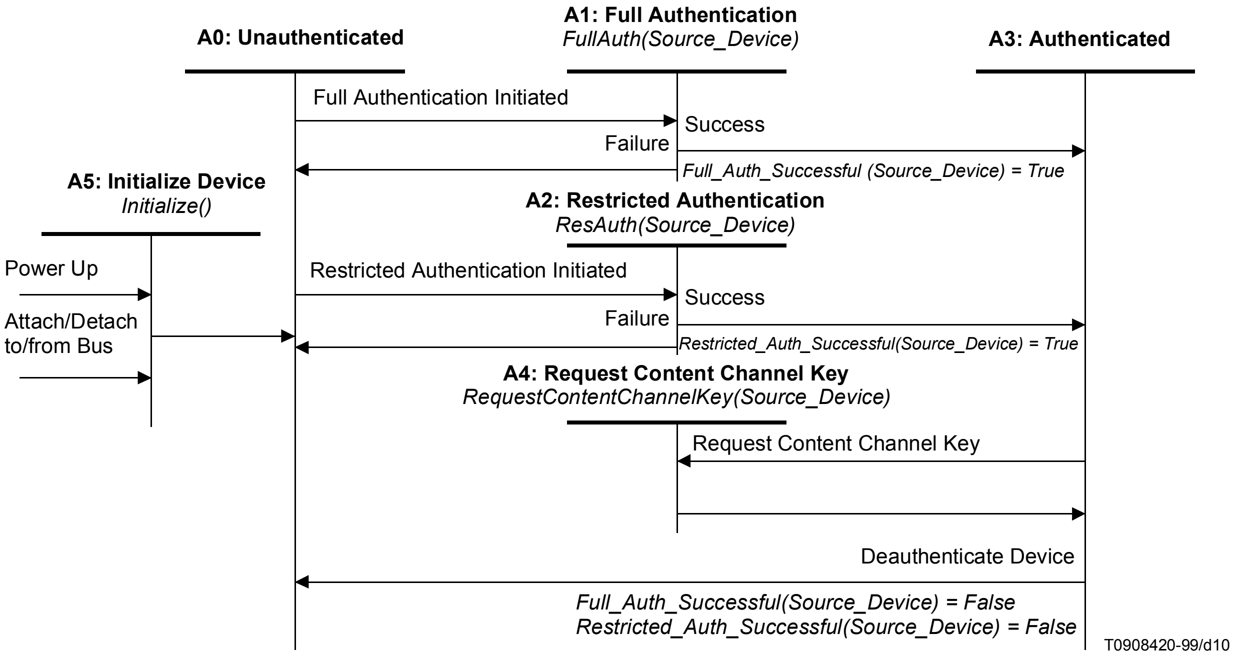
Transition A3:A0.

Set Full_Auth_Successful(Sink_Device) = False

Set Restricted_Auth_Successful(Sink_Device) = False

III.3.2 Content Sink Device

Figure III.6 shows the various states of operation of a device that is a sink for content.



A1: Full Authentication
FullAuth(Source_Device)
A0: Unauthenticated
A3: Authenticated
Full Authentication Initiated
Success
Failure
Full_Auth_Successful (Source_Device) = True
A5: Initialize Device
Initialize()
A2: Restricted Authentication
ResAuth(Source_Device)
Power Up
Restricted Authentication Initiated
Success
Attach/Detach
to/from Bus
Failure
Restricted_Auth_Successful(Source_Device) = True
A4: Request Content Channel Key
RequestContentChannelKey(Source_Device)
Request Content Channel Key
Deauthenticate Device
Full_Auth_Successful(Source_Device) = False
Restricted_Auth_Successful(Source_Device) = False
T0908420-99/d10

Figure III.6/J.95 – Content Sink Device State Machine

A Power up or Attach/Detach to/from the bus event resets this state machine into **State A5: Initialize Device.**

State A5: Initialize Device. In this state, the device is initialized.

Transition A5:A0. This transition to **State A0: Unauthenticated** occurs following the completion of the initialization process.

State A0: Unauthenticated. A device is in an unauthenticated state, waiting to initiate a request to perform the Full or Restricted Authentication procedure.

Transition A0:A1. This transition occurs when the device initiates a request to perform the Full Authentication procedure with another device *(Source_Device)*.

State A1: Full Authentication. In this state, the process *FullAuth(Source_Device)* is performed. This process is described in detail in III.4.

Transition A1:A3. This transition occurs when *FullAuth(Source_Device)* has been successfully completed.

Set Full_Auth_Successful(Source_Device) = True

Transition A1:A0. This transition occurs when *FullAuth(Source_Device)* is unsuccessful.

Transition A0:A2. This transition occurs when the device initiates a request to perform the Restricted Authentication procedure with another device (*Source_Device).*

State A2: Restricted Authentication. In this state, the device executes the process *ResAuth (Source_Device).* This procedure is described in detail in III.5.

Transition A2:A3. This transition occurs when *ResAuth(Source_Device)* has been successfully completed.

Set Restricted_Auth_Successful(Source_Device) = True

Transition A2:A0. This transition occurs when *ResAuth(Source_Device)* is unsuccessful.

State A3: Authenticated. When a device is in this state, it has successfully completed either the Full or Restricted Authentication procedure.

Transition A3:A4. An authenticated device need to request a Content Key to gain access to copy protected content.

State A4: Request Content Channel Key. In this state, an authenticated sink device requests the values necessary to create a Content Key by executing the process *RequestContentChannelKey(Source_Device)*. This process is described in III.6.

Transition A4:A3. This transition occurs on completion of the process *RequestContentChannelKey(Source_Device).*

Transition A3:A0.

Set Full_Auth_Successful(Source_Device) = False

Set Restricted_Auth_Successful(Source_Device) = False

III.4 Full Authentication

III.4.1 Introduction

This subclause addresses the particulars of the Full Authentication level of device authentication and key exchange. Full Authentication employs the public key based Elliptic Curve Digital Signature Algorithm (EC-DSA) for signing and verification. It also employs the Elliptic Curve Diffie-Hellman (EC-DH) key exchange algorithm to generate a shared authentication key.

III.4.2 Notation

The notation introduced in this subclause is used to describe the cryptographic processes. All operations in the elliptic curve domain are calculated on an elliptic curve E defined over GF(p).

III.4.2.1 Defined by the DTLA

The following parameters, keys, constants and certificates are generated by the DTLA.

III.4.2.1.1 General

p = A prime number greater than 3.

GF(p) = The finite field of p elements, represented as the integers modulo p.

E = The elliptic curve over the field GF(p) .

a, b = The coefficients defining the elliptic curve E, elements of GF(p).

G = The basepoint for the elliptic curve.

r = The order of G.

L^{-1}, L^{1} = DTLA EC-DSA key pair consists of an EC private key L^{-1} which is an integer in the range [1, r – 1] and an EC public key L^{1} which is a point on E, where $L^{1} = L^{-1}G$.

These constants, with the exception of L^{-1}, are available in the DTCP Specification available under license from the DTLA.

III.4.2.1.2 For Device X

X^{-1}, X^{1} = Device EC-DSA key pair consists of an EC private key X^{-1} which is an integer in the range [1, r – 1] and an EC public key X^{1} which is a point on E, where $X^{1} = X^{-1}G$.

X_{Cert} = A device certificate given to compliant device X by the DTLA and used during the authentication process (See the next subclause for details).

Elliptic curve points consist of the concatenation x-coordinate and y-coordinate, respectively; for an elliptic curve point P = (x_P, y_P) which is not equal to the elliptic curve point at infinity, P = $x_P \| y_P$.

Table III.1/J.95 – Length of keys and elliptic curve parameters generated by DTLA (Full Authentication)

Key and Elliptic curve parameters	Size (bits)
DTLA Public Key (L^{1})	320
Device Private Key (X^{-1})	160
Device Public Key (X^{1})	320
Basepoint (G)	320
Coefficient of elliptic curve polynomial (a, b)	160 (each)
Prime number (p) of finite field GF(p)	160
Order of Basepoint (r)	160

III.4.2.2 Notation used during Full Authentication

The following additional values are generated and used by the devices during Full Authentication:

X_n = Nonce (random challenge generated by RNG_F).

X_k = Random value used in EC-DH key exchange generated by RNG_F in the device (integer in the range [1, r – 1]).

X_V = EC-DH first phase value (X_kG) calculated in the device (point on the elliptic curve E).

X_{SRMV} = Version number of the system renewability message (SRMV) stored by the device (see III.7).

X_{SRMC} = Indicates the number of SRM part(s) which are currently stored in the non-volatile memory of the device. A value of SRMC indicates that the first SRMC+1 generations of SRMs are currently stored by the device (see III.7).

K_{Auth} = Authentication key which is the least significant 96 bits of shared data created through EC-DH key exchange.

Table III.2/J.95 – Length of keys and variables generated by the device (Full Authentication)

Key or variable	Size (bits)
Nonce (random challenge X_n)	128
Random Value for EC-DH (X_k)	160
EC-DH first phase value (X_V)	320
X_{SRMV}	16
X_{SRMC}	4
Authentication key created through EC-DH key exchange (K_{Auth})	96

III.4.2.3 Device Certificate Formats

A device certificate is given to each compliant device by the DTLA. This certificate is stored in the compliant device and used during the authentication process.

III.4.2.3.1 Baseline Format

Figure III.7 shows the baseline device certificate format:

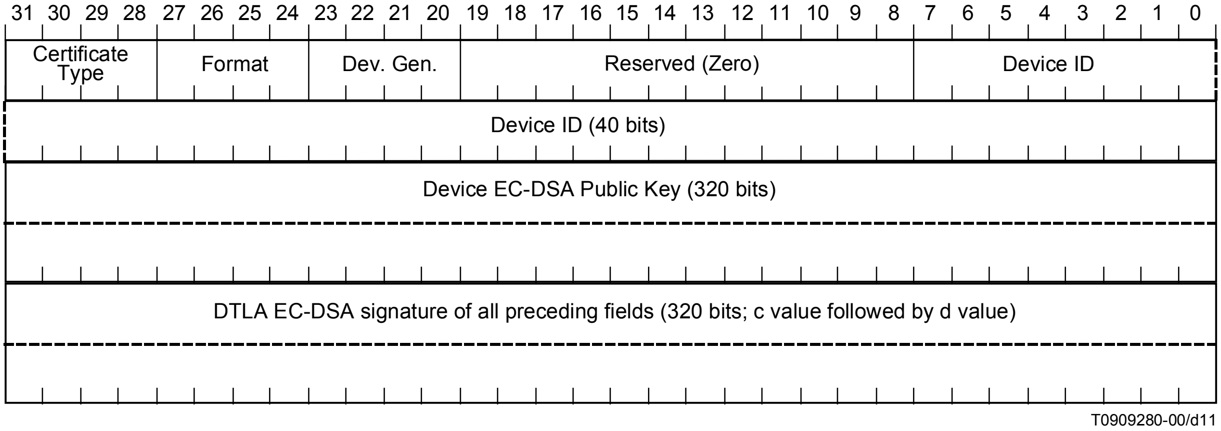


Figure III.7/J.95 – Baseline Device Certificate Format

Device certificates are comprised of the following Baseline Format fields:

- **Certificate Type** (4 bits). The only encoding which is currently defined is 0, which indicates that the certificate is for IEEE 1394 content protection. Other encoding are currently reserved.
- **Certificate Format** (4 bits). This field specifies the format for a specific type of certificate. Currently three formats are defined:

  Format 0 = the Restricted Authentication device certificate format (described in III.5).

  Format 1 = the Baseline Full Authentication device certificate format.

  Format 2 = the Extended Full Authentication device certificate format (Optional[2]).

[2] Features of this specification that are labelled as "optional" describe capabilities whose usage has not yet been established by the DTLA.

Other encodings are currently reserved.

- **Device Generation** (X_{SRMG}) (4 bits). This field indicates the non-volatile memory capacity and therefore the maximum generation of renewability message that this device supports (described in III.7). The encoding 0 indicates a maximum size of 128 bytes while the other encodings are currently reserved.
- Reserved Field (12 bits). These bits are reserved for future definition and are currently defined to have a value of zero.
- The **device's ID** number (X_{ID}, 40 bits) assigned by the DTLA.
- The **EC-DSA public key** of the device (X^1, 320 bits).
- An **EC-DSA signature** from the DTLA of the components listed above (320 bits).

The overall size of a Baseline Format device certificate is 88 bytes.

III.4.2.3.2 Extended Format Fields (Optional Components of the Device Certificate)

In addition to the Baseline Format fields, each device certificate may optionally include the following Extended Format fields[3]:

- A **device capability mask** indicating the properties of the device and channel ciphers supported. (X_{Cap_Mask}, 32 bits)
- A **EC-DSA signature** from the DTLA of all preceding components in the device certificated (320 bits).

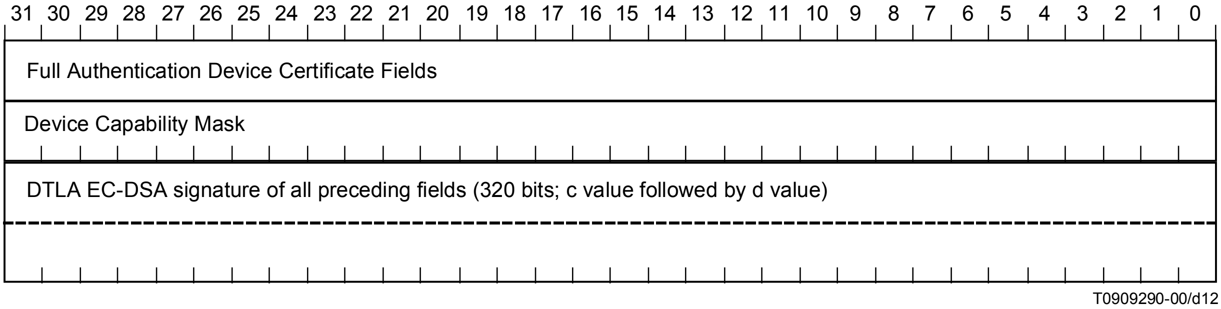

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Full Authentication Device Certificate Fields
Device Capability Mask
DTLA EC-DSA signature of all preceding fields (320 bits; c value followed by d value)
T0909290-00/d12

Figure III.8/J.95 – Extended Device Certificate Fields

Device Capability Mask

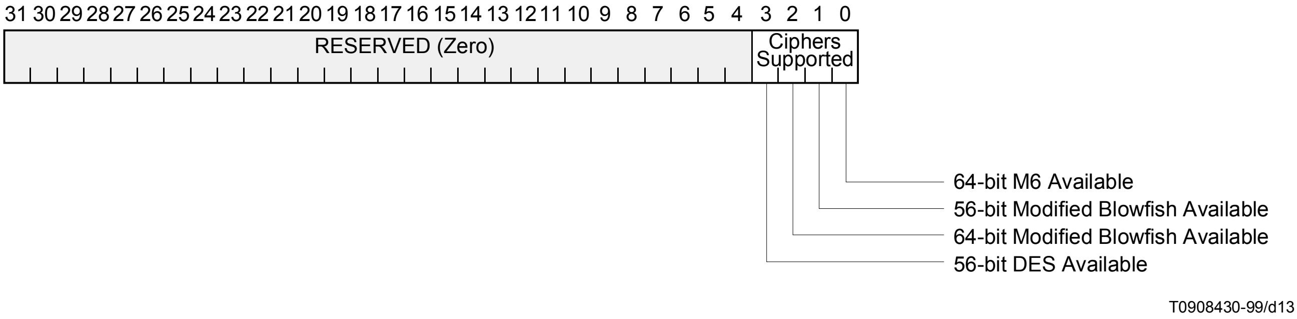


Figure III.9/J.95 – Device Capability Mask

The device capability mask is provided to describe the extensibility features supported by a given device. The format of the mask is shown in Figure III.9.

Devices that do not support the device capability mask are assumed to only support the mandatory cryptographic features defined by this content protection system (e.g. the 56-bit M6 Baseline Cipher).

[3] Features of this specification that are labelled as "optional" describe capabilities whose usage has not yet been established by the DTLA.

III.4.3 Manufacture of compliant devices

All compliant devices that support Full Authentication (that is, each item manufactured, regardless of brand and model) will be assigned a unique device ID (X_{ID}) and device EC-DSA public/private key pair (X^1, X^{-1}) generated by the DTLA. X^{-1} must be stored within the device in such a way as to prevent its disclosure. Compliant devices must also be given a device certificate (X_{Cert}) by the DTLA. This certificate is stored in the compliant device and used during the authentication process. In addition, the compliant device will need to store the other constants and keys necessary to implement the cryptographic protocols.

III.4.4 Cryptographic functions

III.4.4.1 SHA-1 (Secure Hash Algorithm, revision 1)

SHA-1, as described in FIPS PUB 180-1[4] is the algorithm used in DSS to generate a message digest of length 160 bits. A message digest is a value calculated from message. It is similar in concept to a checksum, but computationally infeasible to forge.

III.4.4.2 Random number generator

A high-quality random number generator is required for Full Authentication. The output of this random number generator is indicated by the function RNG_F that is described in the DTCP Specification available under license from the DTLA.

III.4.4.3 Elliptic Curve Cryptography (ECC)

These cryptographic algorithms are based upon cryptographic schemes, primitives, and encoding methods described in IEEE P1363/D3 (11 May, 1998). The IEEE P1363/D3 is an unapproved draft that is subject to change. Changes may occur in subsequent versions of that draft that interfere with conformance to the final IEEE 1363 standard of the cryptographic algorithms described herein.

An Elliptic Curve Cryptosystem (ECC) is used as the cryptographic basis for DH and DSA.

The definition field classifies ECC implementations. For this system, the definition field used is GF(p) where p is a large prime number greater than three. An elliptic curve E over the field GF(p), where $p > 3$, is defined by the parameters a and b and the set of solutions (x, y) to the elliptic curve equation together with an extra point often called the point at infinity. The point at infinity is the identity element of the abelian group, $(E, +)$. The elliptic curve equation used is

$$y^2 = x^3 + ax + b \quad \text{where} \quad 4a^3 + 27b^2 \neq 0$$

where a, b, x, y, are elements of GF(p). A point P on the elliptic curve consists of the x-coordinate and the y-coordinate of a solution to this equation, or the point at infinity, and is designated $P = (x_p, y_p)$.

For EC-DSA and EC-DH, a basepoint G on the elliptic curve is selected. All operations in the elliptic curve domain are calculated on an elliptic curve E defined over GF(p). The public key Y^1 (a point on the elliptic curve) and private key Y^{-1} (a scalar value satisfying $0 < Y^{-1} < r$) for each entity satisfies the equation:

$$Y^1 = Y^{-1} G$$

In specifying the elliptic curve used:

- The order of basepoint G will have a large prime factor.
- The system will be robust against MOV reduction attack, since super singular elliptic curves are avoided.

[4] National Institute of Standards and Technology (NIST), "Secure Hash Standard (SHS)", FIPS Publication 180-1, 17 April, 1995.

III.4.4.3.1 Elliptic Curve Digital Signature Algorithm (EC-DSA)

Signature

The following signature algorithm is based on the ECSSA digital signature scheme using the DLSP-DSA signature primitive and EMSA-SHA-1 encoding method defined in IEEE P1363/D3.

Input:

- M = the data to be signed
- X^{-1} = the private key of the signing device (must be kept secret)
- p, a, b, G and r = the elliptic curve parameters associated with X^{-1}

Output:

- $S_{X^{-1}}[M]$ = a 320-bit signature of the data, M, based on the private key, X^{-1}

Algorithm:

Step 1: Generate a random value, u, satisfying $0 < u < r$, using $\mathbf{RNG_F}$. A new value for u is generated for every signature and shall be unpredictable to an adversary for every signature computation. Also, calculate the elliptic curve point, $V = uG$.

Step 2: Calculate $c = x_V \bmod r$ *(the x-coordinate of V reduced modulo r)*. If $c = 0$, then go to **Step 1**.

Step 3: Calculate $f = [\text{SHA-1}(M)]_{\text{msb\_bits\_in\_r.}}$ That is, calculate the SHA-1 hash of M and then take the most significant bits of the message digest that is the same number of bits as the size of r.

Step 4: Calculate $d = [u^{-1}(f + cX^{-1})] \bmod r$ *(note that u^{-1} is the modular inverse of u mod r while X^{-1} is the private key of the signing device)*. If $d = 0$, then go to **Step 1**.

Step 5: Set first 160 bits of $S_{X^{-1}}[M]$ equal to the big endian representation of c, and the second 160 bits of $S_{X^{-1}}[M]$ equal to the big endian representation of d. ($S_{X^{-1}}[M] = c \parallel d$).

Verification

The following verification algorithm is based on the ECSSA digital signature scheme using the DLVP-DSA signature primitive and EMSA-SHA-1 encoding method defined in of IEEE P1363/D3.

Input:

- $S_{X^{-1}}\ [M]$ = an alleged 320-bit signature ($c \parallel d$) of the data, M, based on the private key, X^{-1}
- M = the data associated with the signature
- X^{1} = the public key of the signing device
- p, a, b, G, and r = the elliptic curve parameters associated with X^{-1}

Output:

- "valid" or "invalid", indicating whether the alleged signature is determined to be valid or invalid, respectively

Algorithm:

Step 1: Set c equal to the first 160 bits of $S_{X^{-1}}[M]$ interpreted as in big endian representation, and d equal to the second 160 bits of $S_{X^{-1}}[M]$ interpreted as in big endian representation. If c is not in the range $[1, r - 1]$ or d is not in the range $[1, r - 1]$, then output "invalid" and stop.

Step 2: Calculate $f = [\text{SHA-1}(M)]_{\text{msb\_bits\_in\_}r}.$ That is, calculate the SHA-1 hash of M and then take the most significant bits of the message digest that is the same number of bits as the size of r.

Step 3: Calculate $h = d^{-1} \bmod r$, $h_1 = (fh) \bmod r$, and $h_2 = (ch) \bmod r$.

Step 4: Calculate the elliptic curve point $P = (x_P, y_P) = h_1G + h_2X^1$. If P equals the elliptic curve point at infinity, then output "invalid" and stop.

Step 5: Calculate $c' = x_P \bmod r$. If $c' = c$, then output "valid"; otherwise, output "invalid".

III.4.4.3.2 Elliptic Curve Diffie-Hellman (EC-DH)

The following shared secret derivation algorithm is based on the ECSVDP-DH primitive defined in IEEE P1363/D3.

Input:

- Y_V = the Diffie-Hellman first phase value generated by the other device (an elliptic curve point)
- p, a, b, G, and r = the elliptic curve parameters associated with X^{-1}

Output:

- X_V = the Diffie-Hellman first phase value (an elliptic curve point)
- the x-coordinate of $X_K Y_V$ = the shared secret generated by this algorithm (must be kept secret from third parties)

Algorithm:

Step 1: Generate a random integer , X_K, in the range [1, r – 1] using **RNG$_\mathbf{F}$**. A new value for X_K is generated for every shared secret and shall be unpredictable to an adversary. Also, calculate the elliptic curve point, $X_V = X_K G$.

Step 2: Output X_V.

Step 3: Calculate $X_K Y_V$. Output the x-coordinate of $X_K Y_V$ as the secret shared.

III.4.4.3.3 Implementation of the Elliptic Curve Cryptosystem

A range of implementation of the Elliptic Curve Cryptosystem can be realized which are compatible with the IEEE P1363 primitives described in this subclause.

An efficient implementation of an elliptic curve cryptosystem can be realized by performing computations within the Montgomery space using new definitions of the basic arithmetic operations of addition, subtraction, multiplication, and inverse[5].

III.4.5 Protocol flow

III.4.5.1 Protocol flow overview

During Full Authentication:

1) The sink device requests authentication by sending a random challenge and its device certificate. This can be the result of the sink device attempting to access a protected content stream (whose EMI is set to "Copy-never", "No-more-copies", or "Copy-one-generation"). The sink device may request authentication on a speculative basis, before attempting to access a content stream. If a sink device attempts speculative authentication, the request can be rejected by the source.

2) Device A then returns a random challenge and its device certificate. If the value of the other device's certificate type or format fields is reserved, the authentication should be immediately aborted. After the random challenge and device certificate exchange, each device verifies the integrity of the other device's certificate using EC-DSA. If the DTLA signature is determined to be valid, the devices examine the certificate revocation list embedded in their system renewability messages (see III.7) to verify that the other device has not been revoked. If the other device has not been revoked, each device calculates a EC-DH key exchange first-phase value (see III.4.4.3.2).

3) The devices then exchange messages containing the EC-DH key exchange first-phase value, the Renewability Message Version Number and Generation of the system renewability message stored by the device, and a message signature containing the other device's random challenge concatenated to the preceding components.

[5] Japanese patent application number: TBD.

The devices verify the signed messages received by checking the message signature using EC-DSA with the other device's public key. This verifies that the message has not been tampered with. If the signature cannot be verified, the device refuses to continue.

In addition, by comparing the exchanged version numbers, devices can at a later time invoke the system renewability mechanisms (see III.7.2 for the details of this procedure).

Each device calculates an authentication key (K_{Auth}) by completing the EC-DH key exchange.

A detailed description of the Full Authentication protocol and associated state machines can be found in the DTCP Specification available under license from the DTLA.

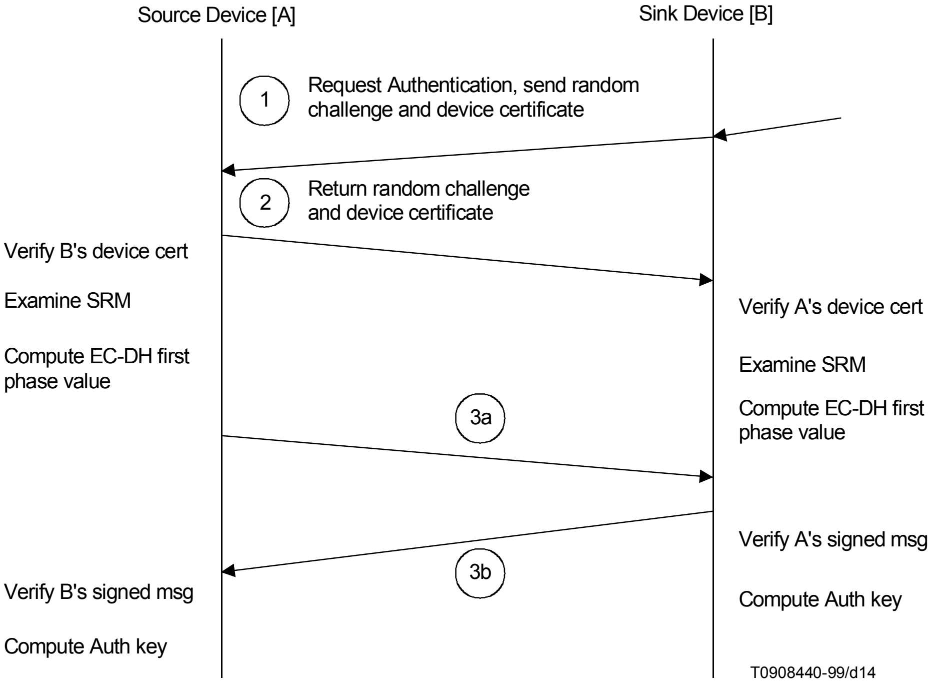


Figure III.10/J.95 – Full Authentication protocol flow overview

III.5 Restricted Authentication

III.5.1 Introduction

This subclause describes the authentication and key exchange between source and sink devices that employ asymmetric key management and common key cryptography for "Copy-one-generation" and "No-more-copies" contents. These kinds of devices, which typically have limited computation resources, follow a Restricted Authentication protocol instead of Full Authentication. Restricted Authentication relies on the use of shared secrets and hash function to respond to a random challenge.

The Restricted Authentication method is based on a device being able to prove that it holds a secret shared with other devices. One device authenticates another by issuing a random challenge that is responded to by modifying it with the shared secret and hashing.

III.5.2 Notation

The notation introduced in this subclause is used to describe the cryptographic process and protocol used for Restricted Authentication.

III.5.2.1 Defined by the DTLA

The following parameters, keys, constants, and certificates must be generated by the DTLA.

III.5.2.1.1 General

The parameters defined in III.4.2.1 are also used during Restricted Authentication by Source devices that also support Full Authentication.

III.5.2.1.2 For Device X

X_{Cert} = A device certificate given to compliant device X by the DTLA and used during the authentication process (See the III.5.2.2 for details).

$X_{Kcosrc1} \ldots X_{Kcosrc12}$ = Each device which is a source of "Copy-one-generation" content receives twelve 64-bit keys from the DTLA.

$X_{Kcosnk1} \ldots X_{Kcosnk12}$ = Each device which is a sink of "Copy-one-generation" content receives twelve 64-bit keys from the DTLA.

$X_{Knmsrc1} \ldots X_{Knmsrc12}$ = Each device which is a source of "No-more-copies" content receives twelve 64-bit keys from the DTLA.

$X_{Knmsnk1} \ldots X_{Knmsnk12}$ = Each device which is a sink of "No-more-copies" content receives twelve 64-bit keys from the DTLA.

X_{KSV} = This key selection vector (KSV) determines which keys will be used during the Restricted Authentication procedure with this device. Only one KSV is required for devices that can be both a source and sink of content.

Table III.3/J.95 – Length of keys and constants created by DTLA (Restricted Authentication)

Key or variable	Size (bits)
"Copy-one-generation" Sink Device Keys ($X_{Kcosnk1} \ldots X_{Kcosnk12}$)	64 (Each)
"Copy-one-generation" Source Device Keys ($X_{Kcosrc1} \ldots X_{Kcosrc12}$)	64 (Each)
"No-more-copies" Sink Device Keys ($X_{Knmsnk1} \ldots X_{Knmsnk12}$)	64 (Each)
"No-more-copies" Source Device Keys ($X_{Knmsrc1} \ldots X_{Knmsrc12}$)	64 (Each)
Key Selection Vector (X_{KSV})	12

Devices contain the keys appropriate to the type of content and functions that they perform.

Notation used during Restricted Authentication

The following additional values are generated and used by the devices during Restricted Authentication:

X_n = Nonce (random challenge generated by RNG_R)

K_V, K'_V = Verification key

R, R′ = Responses to Nonces

K_{Auth}, K'_{Auth} = Authentication key

Table III.4/J.95 – Length of keys and variables generated by the device (Restricted Authentication)

Key or variable	Size (bits)
Nonce (A_n, B_n)	64
Verification Keys (K_v, K'_v)	64
Responses (R, R')	64
Authentication Keys (K_{Auth}, K'_{Auth})	96

III.5.2.2 Device Certificate Format

A Restricted Authentication Device Certificate is used in the Restricted Authentication process. Each Restricted Authentication device certificate is assigned by the DTLA and includes a Device ID and a signature generated by the DTLA. All compliant sink devices that support only Restricted Authentication shall have this certificate.

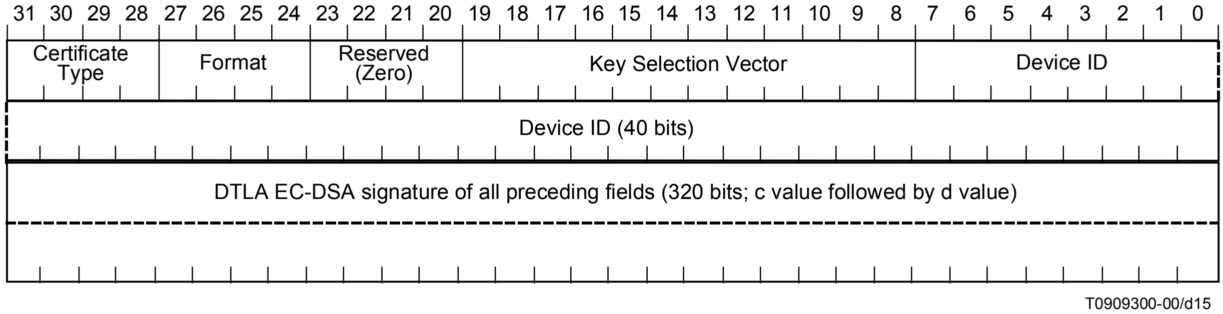


Figure III.11/J.95 – Restricted Authentication device certificate format

The Restricted Authentication device certificate is comprised of the following fields (see Figure III.11) :

- **Certificate Type** (4 bits). (See III.4.2.3.1 for a description of the encoding.)
- **Certificate Format** (4 bits). (See III.4.2.3.1 for a description of the encoding.)
- **Reserved Field** (4 bits). These bits are reserved for future definition and are currently defined to have a value of zero.
- **Key Selection Vector** (X_{KSV}, 12 bits) assigned by the DTLA (see Figure III.12). This vector determines which keys will be used during the Restricted Authentication procedure with this device. This KSV is used regardless of the EMI of the stream to be handled or whether the device is being used as a source or sink of content. The encoding of this field is as follows:
- The **Device ID** number (X_{ID}, 40 bits) assigned by the DTLA.
- A **EC-DSA signature** from the DTLA of the components listed above (320 bits).

The overall size of a Restricted Authentication device certificate format is 48 bytes.

III.5.2.3 Random number generator

A random number generator is required for Restricted Authentication. The output of this random number generator is indicated by the function RNG_R. Either RNG_R or RNG_F as described in the DTCP Specification available under license from the DTLA may be used for Restricted Authentication.

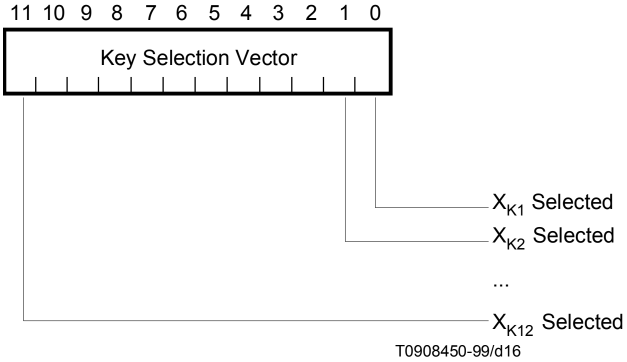


Figure III.12/J.95 – Key Selection Vector

III.5.3 Protocol flow

III.5.3.1 Protocol flow overview

Figure III.13 gives an overview of the Restricted Authentication protocol flow.

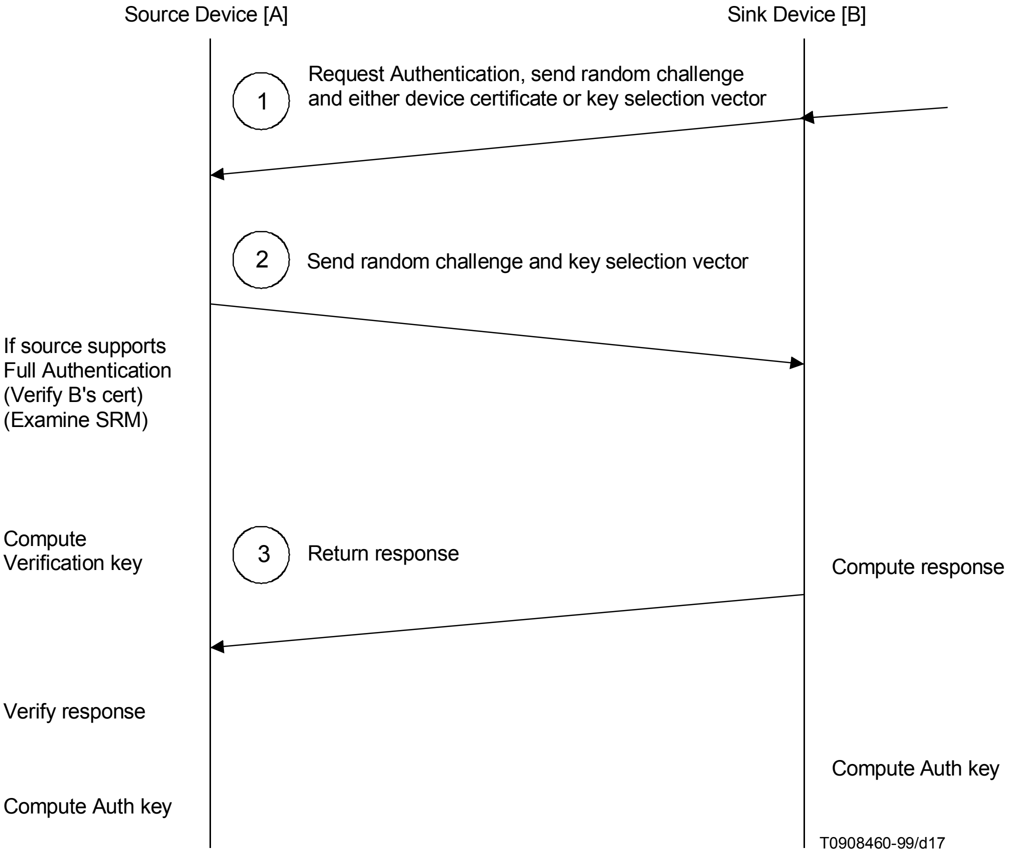

Source Device [A]
Sink Device [B]
1
Request Authentication, send random challenge and either device certificate or key selection vector
2
Send random challenge and key selection vector
If source supports Full Authentication (Verify B's cert) (Examine SRM)
Compute Verification key
3
Return response
Compute response
Verify response
Compute Auth key
Compute Auth key
T0908460-99/d17

Figure III.13/J.95 – Restricted Authentication protocol flow overview

During Restricted Authentication:

1) The sink device initiates the authentication protocol by sending an asynchronous challenge request to the source device. This request contains the type of Exchange Key to be shared by the source and sink devices as well as a random number generated by the sink device using RNG_R. If the sink device knows that the source device does not have a capability for Full Authentication, the sink sends its KSV to the source; otherwise, the sink sends its Restricted Authentication device certificate.

2) The source device generates a random challenge using RNG_R and sends it to the sink device. If the source device supports Full Authentication, it extracts the device ID of the sink device from the certificate sent by the sink. It then checks:

   a) that the certificate sent by the sink is valid; and

   b) that the sink's device ID is not listed in the certification revocation list in the system renewability message stored in the source device.

   Also, if the value of the other device's certificate type or format fields is reserved, the authentication should be immediately aborted. If these checks are completed successfully, the source continues the protocol by computing the verification key.

3) After receiving a random challenge back from the source device, the sink device computes a response using a verification key that it has computed and sends it to the source.

4) After the sink device returns a response, the source device compares this response with similar information generated at the source side using its verification key. If the comparison matches its own calculation, the sink device has been verified and authenticated. If the comparison does not match it, the source device shall reject the sink device. Finally, each device computes the authentication key.

A detailed description of the Restricted Authentication protocol and associated state machines can be found in the DTCP Specification available under license from the DTLA.

III.6 Content Channel management and protection

III.6.1 Introduction

This subclause details the mechanisms used to:

1) share an Exchange Key between a source device and a sink device; and

2) establish and manage the encrypted isochronous channel through which protected content flows.

Either Full or Restricted Authentication (depending on the capabilities of the device) shall be completed prior to establishing a content channel.

III.6.2 Content management keys

III.6.2.1 Exchange Key (K_x)

A common set of Exchange Keys (K_x) are established between a source device and all sink devices that have completed the appropriate authentication procedure (either Full or Restricted) with the source device required to handle content with a specific EMI value (III.6.4.2). In addition, if optional content ciphers[6] are mutually supported, Exchange Keys are established for use with them for Copy-never content.

The procedure for establishing an Exchange Key is described in III.6.3.1.

III.6.2.2 Content Key (K_c)

The **Content Key** (K_c) is used as the key for the content encryption engine. K_c is computed from the three values shown below:

- Exchange Key K_x assigned to the EMI and cipher/key length being used to protect the content.

[6] Only applicable for Exchange Keys established as a result of Full Authentication between devices which both support the optional capability mask in the device certificate.

- A random number N_c generated by the source device (using RNG_F for devices which support Full Authentication and RNG_R for devices which support only Restricted Authentication) which is sent in plain text to all sink devices in asynchronous packet(s).
- Constant value C_a *or* C_b, or C_c, which corresponds to the encryption mode EMI in the packet header.

The Content Key is generated as follows:

$$K_c = J[K_x, N_c, f[\text{EMI}]]$$

where:

$f[\text{EMI}] = C_a$ if EMI is mode A

$f[\text{EMI}] = C_b$ if EMI is mode B

$f[\text{EMI}] = C_c$ if EMI is mode C

C_a, C_b and C_c are universal secret constants assigned by the DTLA. The values for these constants are specified in DTCP Specification available under license from the DTLA. The definition of function $J[]$ is also described in this appendix.

III.6.2.3 Key sizes

Table III.5 lists the lengths of the keys and constants described above.

Table III.5/J.95 – Length of keys and constants (Content Channel management)

Key, Variable, or Constant	Size (bits)
Exchange Keys (K_x)	96
Scrambled Exchange Keys (K_{sx})	96
Constants (C_a, C_b, C_c)	24
Content Key for Baseline Cipher (K_c)	56
Content Key for Optional Ciphers[a)] (K_c)	56-64
Nonce for Content Channel (N_c)	64

[a)] Features of this specification that are labelled as "optional" describe capabilities whose usage has not yet been established by the DTLA.

III.6.3 Protocol flow

III.6.3.1 Establishing Exchange Key(s)

After the completion of Full or Restricted Authentication, the source device establishes the Exchange Key(s) described in III.A.6.2.1. The following procedure is used for each key:

1) The source device assigned a random value for the particular Exchange Key (K_x) being established.

2) It then scrambles the key K_x using K_{Auth} resulting in K_{sx} according to the function described in the DTCP Specification available under license from the DTLA.

3) The source device sends K_{sx} to the sink device.

4) The sink device descrambles the key K_{sx} using K'_{Auth} to determine the shared Exchange Key K_x according to the function described in the DTCP Specification available under license from the DTLA.

The source device repeats the above steps for all of the Exchange Keys required between it and the sink device.

Finally, the devices update the SRM if it is determined to be necessary during the Full Authentication process (see III.4).

Devices remain authenticated as long as they maintain valid Exchange Keys. The Exchange Key may be repeatedly used to set up and manage the security of copyrighted content streams without further authentication. It is recommended that source devices expire their Exchange Keys when they stop all isochronous output. Additionally, devices must expire their Exchange Keys when they are detached from the bus.

III.6.3.2 Establishing Content Keys

This subclause describes the mechanism for establishing the Content Keys ($K_{c)}$ used to encrypt/decrypt content being exchanged between the source and sink devices (see Figure III.14).

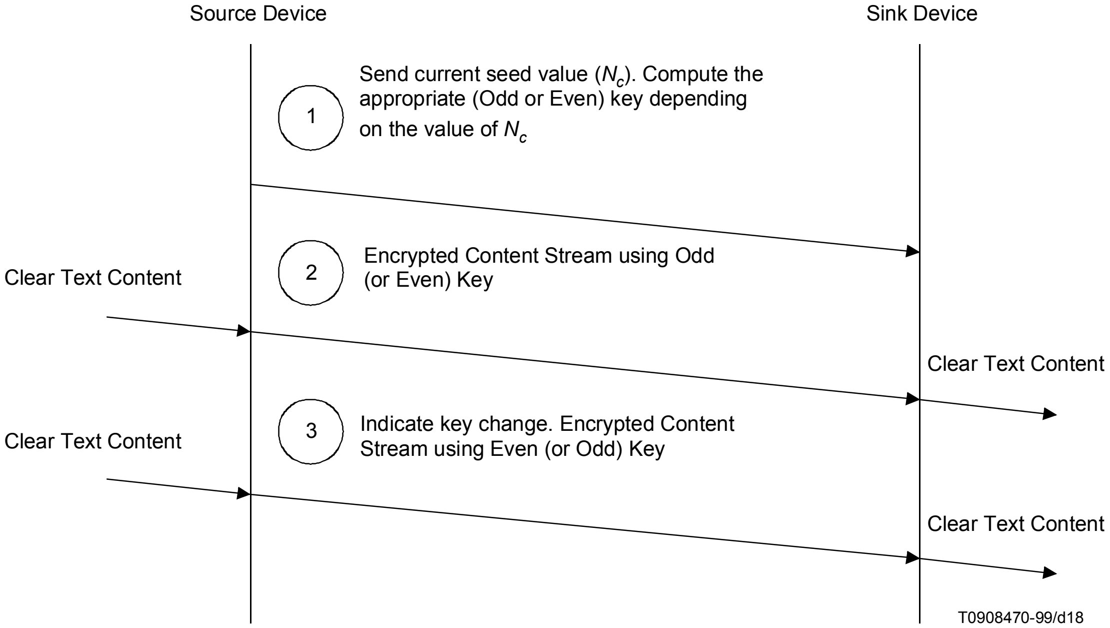

Source Device
Sink Device
1
Send current seed value (N_c). Compute the appropriate (Odd or Even) key depending on the value of N_c
2
Encrypted Content Stream using Odd (or Even) Key
Clear Text Content
Clear Text Content
3
Indicate key change. Encrypted Content Stream using Even (or Odd) Key
Clear Text Content
Clear Text Content
T0908470-99/d18

Figure III.14/J.95 – Content Channel establisment and management protocol flow overview

Content Keys are established between the source device and the sink device as follows:

1) When the source device starts sending content, it generates a 64-bit random number as an initial value of the seed (N_c) of the Content Key. The initial seed is referred to as Odd or Even based on its least significant bit. If subsequent content channels are established, the current value of N_c from the active content channel(s) may be used as the seed.

2) The source device begins transmitting the content using the Odd or Even Content Key (K_c) corresponding to the above reference of the initial seed to encrypt the content. The Content Key is computed by the source device using the function *J*, Exchange Key K_x, the seed (N_c) and the *f[EMI])*. A bit in the IEEE 1394 packet header is used to indicate which key (ODD or EVEN) is being used to encrypt a particular packet of content. If the initial seed is ODD, the Odd/Even bit in the IEEE 1394 packet header is set to Odd; otherwise, it is set to Even.

Upon receiving the seed N_c, the sink device checks if the least significant bit of the N_c matches the status of the Odd/Even bit. If both bits are identical, the sink device computes the current Content Key using the function J, K_x, $f[EMI]$, and N_c. If those bits are different, it shows the key has been changed and the sink device computes the current Content Key by following method:

a) computing $N_c + 1 \ mod \ 2^{64}$ as new seed; then

b) computing the Content Key with above method using the new seed instead of the original seed sent from the source device.

The source device prepares the next Content Key by computing K_c using the same process used for the initial calculation with exception that the seed (N_c) is incremented.

Periodically, the source device shall change Content Keys to maintain robust content protection. To change keys, the source device starts encrypting with the new key computed above and indicates this change by switching the state of the Odd/Even bit in the IEEE 1394 packet header. The minimum period for change of the Content Key is defined as 30 s. The maximum period is defined as 120 s. Duration time for K_c is from 30 s to 2 minutes. A source device should not increment the Content Key duration time counter when it is outputting only contents marked with an EMI value (see III.6.4.2) of Copy-free. When a device suspends all isochronous outputs, it should reset its counter.

The protocol flow to establish the Content Key using IEEE 1394 transactions is shown in III.8.

III.6.3.3 Odd/Even bit

The Odd/Even bit (the third bit of the synch field of the IEEE 1394 isochronous packet header) is used to indicate which Content Key (K_c) is currently being used to protect the content channel (see Figure III.15). The Odd/Even bit only exists when the value of the tag field is 01A; "0" indicates that the Even key should be used while "1" indicates that the Odd key should be used. The Odd key and Even key are used and updated alternately. The Odd/Even bit can only be changed on isochronous packets that contain either the beginning of a new encryption frame or are idle packets between encryption frames. If an isochronous packet contains portions of more than one encryption frame, then the change in key is applied to the first encryption frame which begins in the packet.

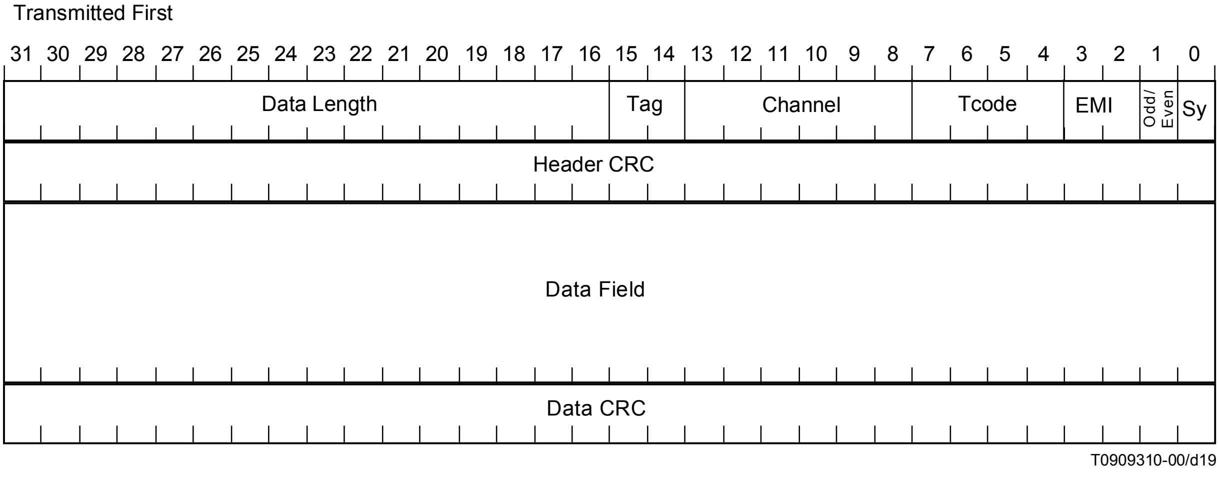


Figure III.15/J.95 – Odd/Even bit location in the packet header

III.6.4 Copy Control Information (CCI)

Copy Control Information (CCI) specifies the attributes of the content with respect to this content protection system. Two CCI mechanisms are supported: Embedded CCI and the Encryption Mode Indicator.

III.6.4.1 Embedded CCI

Embedded CCI is carried as part of the content stream. Many content formats including MPEG have fields allocated for carrying the CCI associated with the stream. The integrity of embedded CCI is ensured since tampering with the content stream results in erroneous decryption of the content.

III.6.4.2 Encryption Mode Indicator (EMI)

The Encryption Mode Indicator (EMI) provides an easy-to-access yet secure mechanism for indicating the CCI associated with a stream of digital content. For IEEE 1394 serial buses, the EMI is placed in the most significant two bits of the Synch field of the packet header as shown in Figure III.16. The EMI bits only exist when the value of the tag field is 01. By locating the EMI in an easy-to-access location, devices can immediately determine the CCI of the content stream without needing to decode the content transport format to extract the embedded CCI. This ability is critical for enabling bit-stream recording devices (e.g. digital VCR) that do not recognize and cannot decode specific content formats.

The EMI bits can only be changed on isochronous packets that contain either the beginning of a new encryption frame or are idle packets between encryption frames. If an isochronous packet contains portions of more than one encryption frame, then the change in EMI is applied to the first encryption frame which begins in the packet.

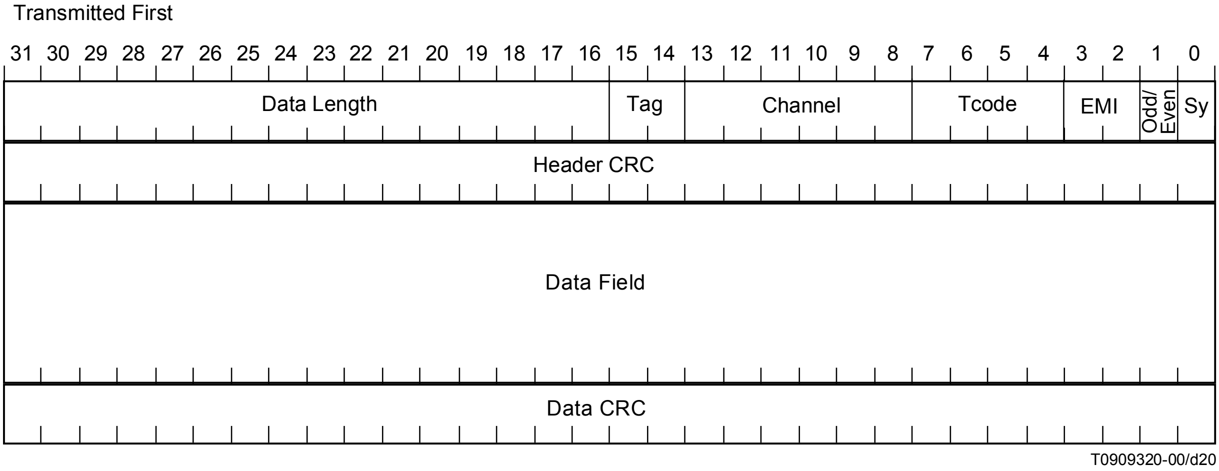


Figure III.16/J.95 – EMI location

The EMI indicates the mode of encryption applied to a stream:

- Licensed source devices will choose the right encryption mode according to the characteristics of the content stream and set its EMI accordingly. If the content stream consists of multiple substreams with different embedded CCI, the strictest embedded CCI will be used to set the EMI.

- Licensed sink devices will choose the right decryption mode as indicated by the EMI.

If the EMI bits are tampered with, the encryption and decryption modes will not match, resulting in an erroneous decryption of the content.

Table III.6/J.95 – EMI encoding

EMI mode	EMI value	Meaning	Authentication required
Mode A	11	Copy-never	Full
Mode B	10	Copy-one-generation	Restricted or Full
Mode C	01	No-more-copies	Restricted or Full
N.A[a]	00	Copy-free	None, not encrypted
[a] Not Applicable. No EMI mode is defined for an encoding of 00.			

- An encoding of 00 is used to indicate that the content can be copied freely. No authentication or encryption is required to protect this content.

- For content that is never to be copied [e.g. content from pre-recorded media with a Copy Generation Management System (CGMS) value of 11], an EMI encoding of 11 is used. This content can only be exchanged between devices that have successfully completed the Full Authentication procedure.

- An EMI encoding of 10 indicates that one generation of copies can be made (e.g. content from prerecorded media with a CGMS value of 10). Devices exchanging this content can either use Full or Restricted Authentication.

- If content with EMI = 10 is copied, future exchanges across a digital interconnect are marked with an EMI encoding of 01, which indicates that a single-generation copy has already been made.

III.6.4.3 Relationship between embedded CCI and EMI

A protected stream of content may consist of one or more programmes. Each of these programmes may be assigned a different level of embedded CCI. Since EMI is associated with the overall stream of content, it is possible that the stream will be composed of multiple programmes and that the EMI will not match the embedded CCI value of each of the protected programmes. In the event that there is a conflict, the most restrictive embedded CCI value will be used for the EMI.

Table III.7/J.95 – Relationship between EMI and embedded CCI

EMI	Embedded CCI for each programme			
	00	01	10	11
Mode A (Copy-never)	Allowed	Allowed[a]	Allowed	Allowed
Mode B (Copy-one-generation)	Allowed	Prohibited	Allowed	Prohibited
Mode C (No-more-copies)	Allowed	Allowed	Allowed	Prohibited
N.A. (Copy-free)	Allowed	Prohibited	Prohibited	Prohibited
[a] Not typically used.				

III.6.4.4 Treatment of EMI/embedded CCI for common device functions

This subclause presents the behaviour of common device functions according to their ability to send/receive EMI and detect/modify embedded CCI. Other functions not listed in this subclause may be permitted as long as they are consistent with the provisions of this specification.

III.6.4.4.1 Format-cognizant source function

A format-cognizant source function (see Table III.8) can recognize the embedded CCI of a content stream being transmitted.

Table III.8/J.95 – Format-cognizant source function CCI handling

Embedded CCI of programmes				EMI
00	**01**	**10**	**11**	
Don't care	(Note)	Don't care	Present	Mode A (Copy-never)
Don't care	Cannot be present	Present	Cannot be present	Mode B (Copy-one-generation)
Don't care	Present	Cannot be present	Cannot be present	Mode C (No-more-copies)
Present	Cannot be present	Cannot be present	Cannot be present	N.A. (Copy-free)
Other combinations				Transmission prohibited
NOTE – Don't care, but not typically used.				

III.6.4.4.2 Format-non-cognizant source function

A format-non-cognizant source function (see Table III.9) need not recognize the embedded CCI of a content stream being transmitted.

Table III.9/J.95 – Format-non-cognizant source function CCI handling

EMI or recorded CCI[a)] of source content	EMI used for transmission
Copy-never	Mode A (Copy-never)
Copy-one-generation	Mode B (Copy-one-generation)
No-more-copies	Mode C (No-more-copies)
Copy-free	N.A. (Copy-free)

[a)] Recorded CCI is copy control information that is not embedded in the content programme and does not require knowledge of the content format to extract.

III.6.4.4.3 Format-cognizant recording function

A format-cognizant recording function (see Table III.10) recognizes the embedded CCI of a received content programme prior to writing it to recordable media.

Table III.10/J.95 – Format-cognizant recording function CCI handling

EMI	Embedded CCI of programme			
	00	01	10	11
Mode A (Copy-never)	Recordable	Do not record	(Note 1)	Do not record
Mode B (Copy-one-generation)	Recordable	Discard entire content stream (Note 2)	(Note 1)	Discard entire content stream (Note 2)
Mode C (No-more-copies)	Recordable	Do not record	Do not record	Discard entire content stream (Note 2)

NOTE 1 – If the recording function supports recording a CCI value of No-more-copies then the CCI value of No-more-copies shall be recorded with the programme. Otherwise, the CCI of Copy-never shall be recorded with the programme.

NOTE 2 – If the function detects this CCI combination among the programmes it is recording, the entire content stream is discarded.

III.6.4.4.4 Format-cognizant sink function

A format-cognizant sink function can recognize the embedded CCI of received content.

Table III.11 shows the embedded CCI of programmes contained within the content stream that can be received.

Table III.11/J.95 – Format-cognizant sink function CCI handling

EMI	Embedded CCI of programme			
	00	01	10	11
Mode A (Copy-never)	Available for processing	Available for processing (Note 2)	Available for processing	Available for processing
Mode B (Copy-one-generation)	Available for processing	Discard entire content stream (Note 1)	Available for processing	Discard entire content stream (Note 1)
Mode C (No-more-copies)	Available for processing	Available for processing	Available for processing (Note 3)	Discard entire content stream (Note 1)

NOTE 1 – If the function detects this CCI combination among the programmes it is recording, the entire content stream is discarded.

NOTE 2 – Not typically used.

NOTE 3 – If the device has a rule for handling No-more-copies, this programme shall be handled according to the rule. Otherwise the programme shall be handled as Never-copy.

III.6.4.4.5 Format-non-cognizant recording function

A Format-non-cognizant recording function (see Table III.12) can record content with appropriate EMI onto recordable media.

Table III.12/J.95 – Format-non-cognizant recording function CCI handling

EMI of the received stream	Recorded CCI[a)] to be written onto user recordable media
Mode A (Copy-never)	Stream cannot be recorded
Mode B (Copy-one-generation)	No-more-copies
Mode C (No-more-copies)	Stream cannot be recorded
[a)] Recorded CCI is copy control information that is not embedded in the content programme and does not require knowledge of the content format to extract.	

III.6.4.4.6 Format-non-cognizant sink function

For this function, the content must be treated in a manner consistent with its EMI 6.5 Common Device Categories.

Devices may support zero or more of the functions described in III.6.4.4.

Common types of fixed function devices include, but are not limited to the following:

1) **Format-cognizant pre-recorded content source device** has a format-cognizant source function (e.g. DVD player).

2) **Format-cognizant real-time-delivery content source/decoding device** has a format-cognizant source function and format-cognizant sink function (e.g. set-top box or digital TV).

3) **Format-cognizant recorder and player** has a format-cognizant source function, format-cognizant sink function, and format-cognizant recording function (e.g. DV-VCR).

4) **Format-non-cognizant recorder and player** has a format-non-cognizant source function and format-non-cognizant recording function (e.g. D-VHS VCR).

5) **Format-non-cognizant Bus Bridge** has a format-non-cognizant source function and format-non-cognizant sink function (e.g. IEEE 1394 to IEEE 1394 bus bridge).

III.6.5 Content channel ciphers

All compliant devices support the baseline cipher and possibly additional, optional ciphers for protecting content[7].

III.6.5.1 Baseline cipher

All devices and applications must, at a minimum, support the baseline cipher to ensure interoperability. The M6-S56 block cipher using the converted cipher-block-chaining (C-CBC) mode is the baseline cipher. This cipher is described in detail in the DTCP Specification available under license from the DTLA.

[7] Features of this specification that are labelled as "optional" describe capabilities whose usage has not yet been established by the DTLA.

III.6.5.2 Content encryption formats

Table III.13 shows the content encryption formats that will be used with content channel ciphers.

Table III.13/J.95 – Content Encryption Formats

Data format	Encryption frame	Size
MPEG Transport Stream	IEC 61883-4 Transport Stream Packet	188 bytes
DV (SD Format)	IEC 61883-2 Isochronous Transfer Unit	480 bytes
Audio	IEC 61883-6 (IEC-PAS) IEC 958 conformant data for 2 channels	8 bytes

III.6.5.3 Support for Optional Content Channel Ciphers

Support is defined in III.4 (Device Capability Mask), Section A.6 of (Establishment of multiple K_X values), Section A.8 of (Encoding of cipher selection in the AV/C Digital Interface Command Set). Optional content channel ciphers algorithms with the converted cipher-block-chaining (C-CBC) mode are described in the DTCP Specification available under license from the DTLA[8].

III.7 System renewability

III.7.1 Introduction

Compliant devices that support Full Authentication can receive and process system renewability messages (SRMs) created by the DTLA and distributed with content. These messages are used to ensure the long-term integrity of the system.

III.7.1.1 SRM message components and layout

There are several components to a system renewability message (SRM):

- A message **Type** field (4 bits). This field has the same encoding as is used for the certificate type field in device certificates. See III.A.4.2.3.1 for a description. The only encoding currently defined is 0, which indicates that the message is for IEEE 1394 content protection.
- A message **Generation** field (SRMM) (4 bits). This field specifies the generation of the SRM. It is used to ensure the extensibility of the SRM mechanism. Currently, the only encoding defined is 0, indicating a first generation SRM with a maximum size as specified in the DTCP Specification available under license from the DTLA. Other encodings are currently reserved. This value remains unchanged even if only part of the SRM can be stored by the device (e.g. X_{SRMC} <= SRMM).
- Reserved field (8 bits). These bits are reserved for future definition and are currently defined to have a value of zero.
- A monotonically increasing system renewability message Version Number (SRMV) (16 bits). This value is exchanged as X_{SRMV} during Full Authentication. This value is not reset to zero when the message generation field is changed.
- Certificate Revocation List (CRL) Length (16 bits). This field specifies the size (in bytes) of the CRL including the CRL Length field (2 bytes), CRL Entries (variable length), and DTLA Signature (40 bytes).
- CRL Entries (variable sized). The CRL used to revoke the certificates of devices whose security has been compromised. Its format is described in the following subclause.
- The DTLA EC-DSA signature of these components using L^{-1} (320 bits).

[8] Features of this specification that are labelled as "optional" describe capabilities whose usage has not yet been established by the DTLA.

The structure of first-generation SRMs is shown in Figure III.17. The fields in the first 4 bytes of the SRM comprise the SRM Header.

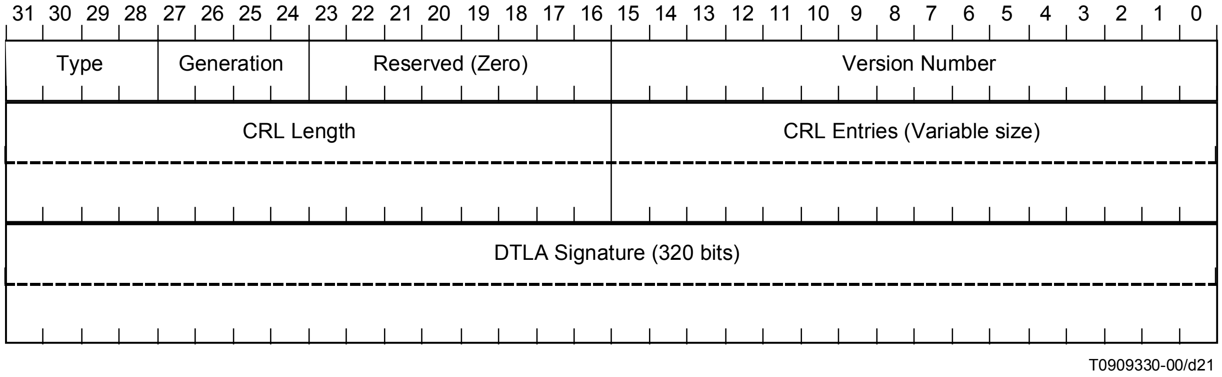


Figure III.17/J.95 – Structure of the first generation System Renewability Message

III.7.1.1.1 Certificate Revocation List (CRL)

The **Certificate Revocation List (CRL)** identifies devices that are no longer compliant. It consists of the CRL Length field that specifies the length of the CRL in bytes. This field is followed by a sequence of entry type blocks (1 byte) which are in turn followed by the number of CRL entries specified by the entry type block. Two types of entry block are supported. One type provides for the revocation of individual devices while the second allows for the revocation of blocks of up to 65 535 devices.

III.7.1.1.2 DTLA EC-DSA Signature

The DTLA EC-DSA Signature field is a 320-bit signature calculated over all of the preceding fields of the SRM using the DTLA EC-DSA private key L^{-1}. This field is used to verify the integrity of the SRM using the DTLA EC-DSA public key L^{-1}.

III.7.1.2 SRM scalability

To ensure the scalability of this renewability solution, the SRM format is extensible (see Figure III.18). Next-generation extensions (CRLs and possibly other mechanisms) to a current-generation SRM format must be appended to the current-generation SRM in order to ensure backward compatibility with devices that only support previous-generation SRMs. Devices are only responsible for supporting the generation of SRM that was required by the DTLA as of the time the device was manufactured. The conditions under which the DTLA will authorize a new-generation SRMs are specified in the DTLA license agreement.

III.7.2 Updating SRMs

System renewability messages can be updated from:

- other compliant devices (connected via the digital transmission means) that have a newer list;
- pre-recorded content media;
- content streams via real-time compliant devices that can communicate externally (e.g. via the Internet, phone line, cable system, direct broadcast satellite, etc.)

The general procedure for updating SRMs is as follows:

1) Examine the version number of the new SRM.
2) Verify that the SRM version number is greater than the one stored in non-volatile storage.
3) Verify integrity with the DTLA public key (L^1).
4) If SRM is valid and new, then store as much as will fit of the newer version of the message in the device's non-volatile storage.

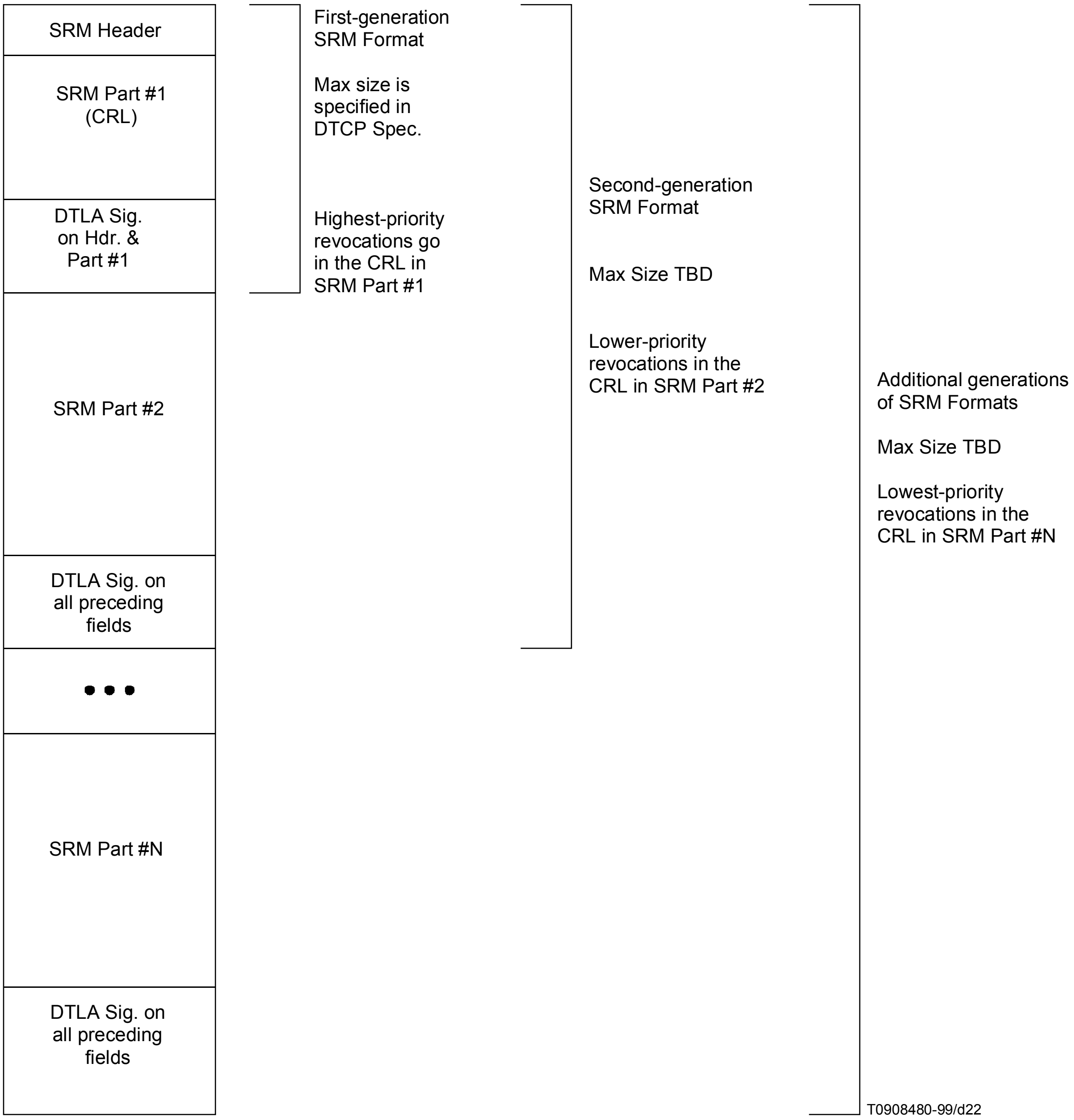

SRM Header
SRM Part #1 (CRL)
DTLA Sig. on Hdr. & Part #1
SRM Part #2
DTLA Sig. on all preceding fields
• • •
SRM Part #N
DTLA Sig. on all preceding fields
First-generation SRM Format
Max size is specified in DTCP Spec.
Highest-priority revocations go in the CRL in SRM Part #1
Second-generation SRM Format
Max Size TBD
Lower-priority revocations in the CRL in SRM Part #2
Additional generations of SRM Formats
Max Size TBD
Lowest-priority revocations in the CRL in SRM Part #N
T0908480-99/d22

Figure III.18/J.95 – SRM extensibility

III.8 AV/C Digital Interface Command Set Extensions

III.8.1 Introduction

Audio/video devices which exchange content via the IEEE 1394 serial bus are typically IEC 61883 and AV/C Digital Interface Command Set compliant. It is important to review Sections A.5, A.6 and A.7 of the *Specification for AV/C Digital Interface Command Set* (General Specification) for general rules about the AV/C commands and responses.

These specifications define the use of IEEE 1394 asynchronous packets for the control and management of devices and IEEE 1394 isochronous packets for the exchange of content. This chapter describes extensions to the AV/C command set which support the DTCP authentication and key exchange protocols. Extensions to the IEEE 1394 Isochronous packet format are described in Section A.6.

III.8.2 SECURITY command

A new Security command is defined for AV/C. This command is intended for content protection purposes including the DTCP system. The general format of the SECURITY command is as follows:

	msb	lsb
Opcode	SECURITY ($0F_{16}$)	
Operand[0]	category	(msb)
Operand[1]	category dependent field	
:		
Operand[X]		(lsb)

The value of the Security Command opcode is $0F_{16}$ (Common Unit and Subunit command).

The **category** field for the SECURITY command is defined as follows:

Value	**category**
0	Support for DTCP AKE. This command is called the AKE command.
1-D_{16}	Reserved for future extension
E_{16}	Vendor_dependent
F_{16}	Extension of category field

The value 0 of the category field specifies that this command is used to support the DTCP Authentication and Key Exchange protocols.

The AKE command is defined for the *ctype* of CONTROL and STATUS. Devices that support the AKE command shall support both *ctype*s.

The value E_{16} of the category field specifies that this command is used by vendors to specify their own security commands for licensed use.

III.8.3 AKE command

The destination of this command is the target device itself. Therefore the 5-bit subunit_type field of an AV/C command/response frame is equal to 11111_2 and the 3-bit subunit_ID field of the frame is equal to 111_2.

III.8.3.1 AKE control command

The AKE control command is used to exchange the messages required to implement the Authentication and Key Exchange protocols. The format of this command is shown below:

Both the AKE Command and Response frames have the same opcode and first 9 operands (operand[0-8]). The value of each field in the response frame is identical to that of the command frame except for the status and data fields. If any of the fields in the first 9 operands contain reserved values, a response of NOT_IMPLEMENTED should be returned.

If a given command frame includes a data field, the corresponding response frame does not have a data field. AKE control commands are used to send the information used for the authentication procedure being performed between the source and sink devices. This information is sent in the data field and is called AKE Info. Non-zero values in Reserved_zero fields of AKE Info should be ignored.

The AKE_ID field specifies the format of the AKE_ID dependent field. Currently, only the encoding AKE_ID = 0 is defined. The AKE_ID dependent field for this encoding is described in III.8.3.3. The other values, from 1_{16} to F_{16}, are reserved for future definition.

	msb	lsb
Opcode	$0F_{16}$	
Operand[0]	category = 0000_2 (AKE)	AKE_ID
Operand[1]	(msb) AKE_ID dependent field	
Operand[2]		
Operand[3]		
Operand[4]		(lsb)
Operand[5]	AKE_label	
Operand[6]	number (option)	status
Operand[7]	blocks_remaining	(msb)
Operand[8]		data_length (lsb)
Operand[9]	data	
:		
Operand[8+ data_length]		

The AKE_label field is a unique tag which is used to distinguish a sequence of AKE commands associated with a given authentication process. The initiator of an authentication procedure can select an arbitrary value for the AKE_label. The value selected should be different from other AKE_label values that are currently in use by the device initiating the authentication. The same AKE_label value will be used for all control commands associated with a specific authentication procedure between a source and sink device. The AKE_label and source node ID of each control command should be verified to ensure that it is from the appropriate controller.

The optional number field[9] specifies the step number of a specific control command to identify its position in the sequence of control commands making up an authentication procedure. The initiator of an authentication procedure sets the value of this field to 1 for the initial AKE control command. The value is incremented for each subsequent command that is part of the same authentication process. When an AKE command must be fragmented for transmission (see the description of the blocks_remaining field below), each fragment will use the same value for the number field. Devices that do no not support this field shall set its value to 0000_2.

The status field is used to notify the device issuing the command of the reason when the command results in a REJECTED response. The device issuing the command sets the value of this field to 1111_2. If the responding device rejects the command, it overwrites the status field with a code indicating the reason for rejection. The encoding of the status field is as follows:

Value	Status	Response code
0000_2	No error	ACCEPTED
0001_2	Support for no more authentication procedures is currently available	REJECTED
0010_2	No isochronous output	REJECTED
0011_2	No point to point connection	REJECTED
0111_2	Any other error	REJECTED
1111_2	No information	Reserved for INTERIM[a)]
[a)] Reserved for future use. Response with INTERIM response code should not currently be used.		

[9] Features of this specification that are labelled as "optional" describe capabilities whose usage has not yet been established by the DTLA.

The following status codes are for testing purposes only. Products shall not return these codes, but instead return 0111_2 (any other error) if these conditions occur.

Value	Status	Response code
1000_2	Incorrect command order (only for test)	REJECTED
1001_2	Authentication failed (only for test)	REJECTED
1010_2	Data field syntax error (only for test)	REJECTED

The blocks_remaining field is used when a command is larger than the maximum command frame size that the target device can receive (A device issuing a command can determine the size of data field that the target device can accept using the AKE status command). When this occurs, the data field is fragmented into N blocks that are sent sequentially, each in one of N separate commands, where each command is small enough to be accommodated by the target device's command buffer. At a minimum, the buffer must be able to hold a command with at least a 32-byte data field[10]. The size of the data field in the first N – 1 fragments shall be the same size and a multiple of 16 bytes greater than or equal to 32 bytes.

Each of the N command frames is identical except for the values in the blocks_remaining, data_length, and data fields. For the first command, the blocks_remaining field is set to the value of N-1. In each successive command, the blocks_remaining field is decreased by one until it reaches zero, indicating the last command fragment. If the value of the block_remaining field is not correct (e.g. not in the correct order), the target should return a REJECTED response with status field of 0111_2 (Any other error).

Since the size of the command and response frames cannot exceed the 512-byte limit imposed by the underlying FCP transport, the case where a command must be fragmented can only occur when a target device has a command frame buffer capacity less than 512 bytes. Typically the command's size is within the target device's command frame buffer capacity and the command is sent without fragmentation and with a blocks_remaining value of zero.

When an AKE_Info is transmitted using multiple Control Commands, a controller shall send each command only after receiving an ACCEPTED response for the previous command.

The data_length field specifies the length of data field in bytes. Responses to a command will use the same value for their respective data_length fields even when the response returns no data. If a response has some data when the response code is ACCEPTED, the corresponding command will have no data but the value of the data_length field shall be the same as that of response.

The data field contains the data to be transferred. The contents of the data field depend on the AKE_ID field and the AKE_ID dependent field. For responses with a response code of REJECTED, there is no data field.

[10] If future generations of System Renewability Messages (SRMM > 0) are defined which have a maximum size larger than 4096 bytes, new devices will be required to support an increase in the minimum buffer size.

III.8.3.2 AKE status command

The format of the AKE status command is as follows:

<table>
<tr><td></td><td>msb</td><td></td><td>lsb</td></tr>
<tr><td>Opcode</td><td colspan="3">$0F_{16}$</td></tr>
<tr><td>Operand[0]</td><td>category = 0000_2 (AKE)</td><td colspan="2">AKE_ID</td></tr>
<tr><td>Operand[1]</td><td colspan="3" rowspan="4">(msb)
AKE_ID dependent field
(lsb)</td></tr>
<tr><td>Operand[2]</td></tr>
<tr><td>Operand[3]</td></tr>
<tr><td>Operand[4]</td></tr>
<tr><td>Operand[5]</td><td colspan="3">FF_{16}</td></tr>
<tr><td>Operand[6]</td><td>F_{16}</td><td colspan="2">status</td></tr>
<tr><td>Operand[7]</td><td colspan="2">$7F_{16}$</td><td>(msb)</td></tr>
<tr><td>Operand[8]</td><td colspan="3">data_length (lsb)</td></tr>
</table>

Both the Command and Response frames have the same structure. The values of each field of the command and response frames are identical except for the AKE_ID dependent, status, and data_length fields.

The AKE_ID field specifies the format of the AKE_ID dependent field. The AKE_ID dependent field for this encoding will be described in III.8.3.3. Currently, only the encoding of AKE_ID = 0 is defined. The other values, from 1_{16} to F_{16} are reserved for future definition.

The status field is used by a device to query the state of another device. When the command is issued, the value of this field is set to 1111_2. In the response, the target device overwrites this field with a value indicating its current situation.

Value	Status	Response code
0000_2	No error	STABLE
0001_2	Support for no more authentication procedures is currently available	STABLE
0010_2	No isochronous output	STABLE
0011_2	No point to point connection	STABLE
0111_2	Any other error	STABLE
1111_2	No information[a)]	REJECTED

[a)] It is recommended that implementers not use the "No information" response.

The following status codes are for testing purposes only. Products shall not return these codes, but instead return 0111_2 (any other error) if these conditions occur.

Value	Status	Response code
1001_2	Authentication failed (only for test)	STABLE

The data_length field specifies the target device's maximum data field capacity in bytes. When the status command is issued, the value of this field is set to $1FF_{16}$. In the response, the target device overwrites this field with a value indicating its current situation. The minimum value to be supported is 020_{16} (32 bytes).

III.8.3.3 AKE_ID dependent field (AKE_ID = 0)

When AKE_ID = 0, the format of the AKE_ID dependent field is as follows:

	msb lsb
Operand[1]	subfunction
Operand[2]	AKE_procedure
Operand[3]	exchange_key
Operand[4]	subfunction_dependent

The subfunction field specifies the operation of control commands. The most significant bit of the subfunction field indicates whether the control command has data or not.

- If the *MSB* is 0, that command has some data and the data_length field indicates its length.
- If the *MSB* is 1, that command has no data and the data_length field indicates the length of the data field in response frame whose response code is ACCEPTED.

The subfunctions are fully described in the DTCP Specification available under license from the DTLA. The following table summarizes the six subfunctions that are currently defined:

Value	Subfunction	Comments
01_{16}	CHALLENGE	Send random value. This subfunction when sent from a sink device initiates the AKE procedure.
02_{16}	RESPONSE	Return data computed with the received random value.
03_{16}	EXCHANGE_KEY	Send an encrypted Exchange Key (K_x) to the authenticated contents-sink device.
04_{16}	SRM	Send SRM to a device that has an outdated or smaller SRM.
$C0_{16}$	AKE_CANCEL	Notify a device that the current authentication procedure cannot be continued.
80_{16}	CONTENT_KEY_REQ	Request the data required for making Content Key (K_c).

For status commands, the value of the subfunction field shall be set to FF_{16}.

Each bit of the AKE_procedure field corresponds to one type of authentication procedure, as described in the table below.

Bit	AKE_procedure
0 (lsb)	Restricted Authentication procedure (rest_auth)
1	Enhanced Restricted Authentication procedure (en_rest_auth) (Note 1)
2	Full Authentication procedure (full_auth)
3	Extended Full Authentication procedure (Note 2) (ex_full_auth, optional) (Note 3)
4-7 (msb)	Reserved for future extension and shall be zero
NOTE 1 – Source devices that support the Full Authentication procedure shall verify the device certificate of the sink device and examine the SRM even in Restricted Authentication. This authentication procedure is referred to as Enhanced Restricted Authentication in this subclause. NOTE 2 – Devices that support extended device certificates use the Extended Full Authentication procedure described in this subclause. NOTE 3 – Features of this specification that are labelled as "optional" describe capabilities whose usage has not yet been established by the DTLA.	

For the control command, the initiator of an authentication procedure sets one bit in this field to specify which type of authentication will be performed. The value of the field then remains constant through the rest of that authentication procedure.

For the status command, the initiator shall set the initial value of this field to FF_{16}. The target will overwrite the field, clearing the bits that indicate the authentication procedures that the target does not support as a source device. For example, if a source device supports both Full Authentication and Enhanced Restricted Authentication, the values of the AKE_procedure field would be set to 06_{16}.

Sink devices should investigate which authentication procedures a source device supports using the status command prior to starting the authentication protocol. The following table shows how to select the appropriate authentication procedure:

Authentication Procedure Supported by the Sink Device Authentication Procedure Supported by the Source Device	Rest_auth and En_rest_auth	Rest_auth and Full_auth	Rest_auth, Full_auth, and Ex_full_auth
Rest_auth	Restricted Authentication	Restricted Authentication	Restricted Authentication
En_rest_auth and Full_auth	Enhanced Restricted Authentication	Full Authentication	Full Authentication
En_rest_auth, Full_auth, and Ex_full_auth	Enhanced Restricted Authentication	Full Authentication	Extended Full Authentication

Each bit of the exchange_key field corresponds to one (or more) key(s) as described in the table below:

Bit	exchange_key
0 (lsb)	Exchange Key(s) for Copy-never content [requires Full or Extended Full Authentication (Note)]
1	Exchange Key for Copy-one-generation content (any authentication acceptable)
2	Exchange Key for No-more-copies content (any authentication acceptable)
3-7 (msb)	Reserved for future extension and shall be zero
NOTE – If Extended Full Authentication is used, all Exchange Keys for mutually support optional ciphers will be sent following the completion of Full Authentication.	

For the control command, the sink device sets the value of this field at the start of an authentication procedure to specify which Exchange Key(s) will be supplied by the source device after the successful completion of the procedure. For Full Authentication any bit can be set. For Restricted Authentication, only one bit for Copy-one-generation or No-more-copies shall be set. This field remains constant for the remainder of the authentication procedure except when the EXCHANGE_KEY subfunction is performed.

For the status command, the initiator shall set FF_{16} in this field, and target shall clear every bit of the field that corresponds to an Exchange Key that the target cannot supply.

For example, if target can supply three keys that correspond to bit0 through bit2 in the table above, the value of the exchange_key field will be set to 07_{16}.

A sink device should decide which key(s) it will require by getting this information in advance of the authentication procedure.

The definition of the subfunction_dependent field varies. The DTCP Specification available under license from the DTLA describes the definitions for control commands. For status commands the value of this field is set to FF_{16} for both the command and response frames.

III.8.4 Bus reset behaviour

If the source device continues to transmit content on an isochronous channel following a bus reset, the same Exchange Keys and Content Keys shall be used as were in use prior to the reset.

If a bus reset occurs during an authentication procedure, both the source and sink devices shall immediately stop the authentication procedure. Following the reset, the Source Node ID (SID) field in the CIP header may have changed requiring the sink device to restart the authentication procedure using the new SID.

III.8.5 Action when unauthorized device is detected during authentication

After returning an ACCEPTED response to an initiator of a command, the target examines the AKE information. If the target determines that the initiator is an unauthorized device, then the target shall immediately stop the AKE procedure without any notification.

III.8.6 Authentication AV/C command flows

Figures III.19 and III.20 illustrate the AV/C command flows used for Full and Enhanced Restricted/Restricted Authentication.

III.8.6.1 Figure notation

Solid lines indicate command/response pairs that are always performed.

Dashed lines indicate command/response pairs that are performed on a conditional basis.

III.8.6.2 Full Authentication command flow

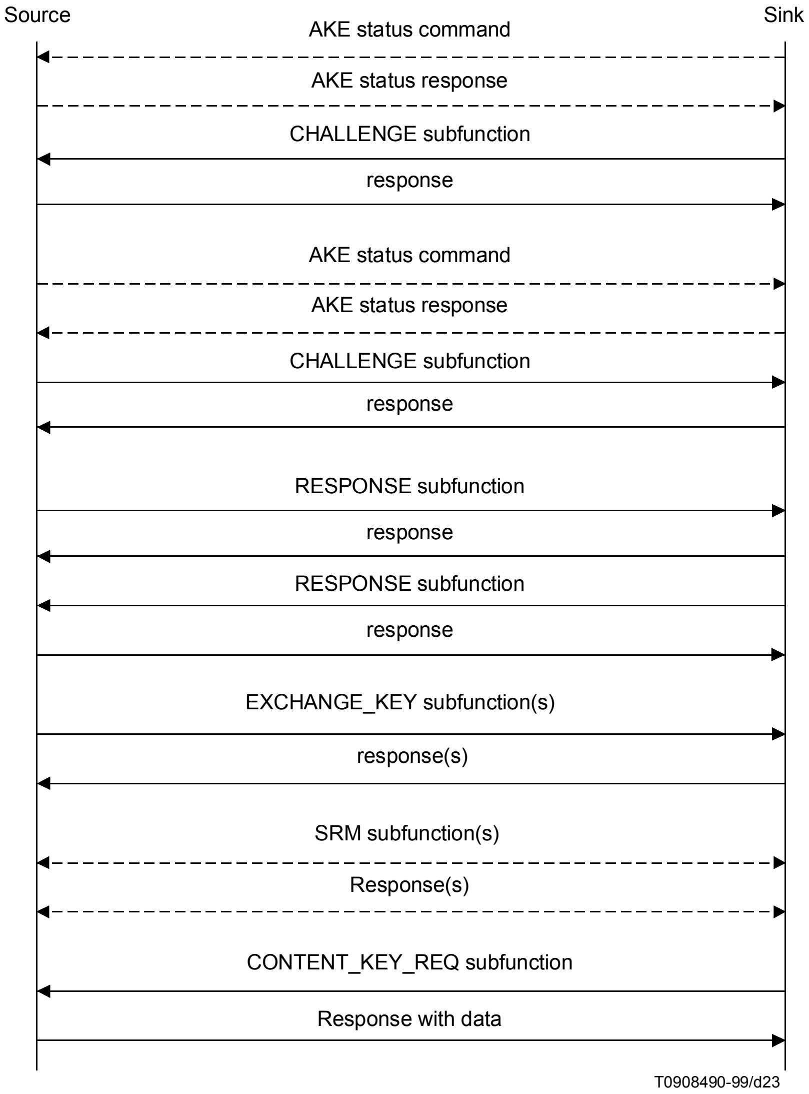


Figure III.19/J.95 – Full Authentification command flow

III.8.6.3 Enhanced restricted/Restricted Authentication command flow

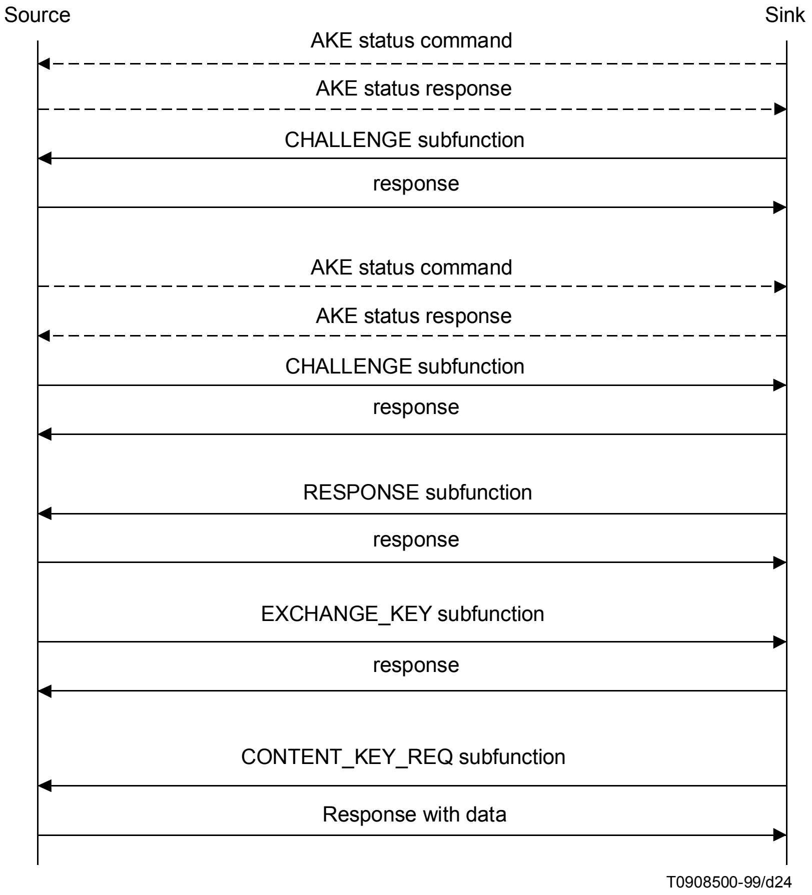


Figure III.20/J.95 –Enhanced Restricted/Restricted Authentication command flow

ITU-T RECOMMENDATIONS SERIES

Series A	Organization of the work of the ITU-T
Series B	Means of expression: definitions, symbols, classification
Series C	General telecommunication statistics
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	**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	TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits
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 and open system communications
Series Y	Global information infrastructure
Series Z	Languages and general software aspects for telecommunication systems

Geneva, 2001