

International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.1194

(04/2012)

SERIES X: DATA NETWORKS, OPEN SYSTEM
COMMUNICATIONS AND SECURITY

Secure applications and services – IPTV security

Algorithm selection scheme for service and content protection descrambling

Recommendation ITU-T X.1194



International
Telecommunication
Union

Documents
Document Sharing Hub

ITU-T X-SERIES RECOMMENDATIONS
DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

PUBLIC DATA NETWORKS	X.1–X.199
OPEN SYSTEMS INTERCONNECTION	X.200–X.299
INTERWORKING BETWEEN NETWORKS	X.300–X.399
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	X.500–X.599
OSI NETWORKING AND SYSTEM ASPECTS	X.600–X.699
OSI MANAGEMENT	X.700–X.799
SECURITY	X.800–X.849
OSI APPLICATIONS	X.850–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999
INFORMATION AND NETWORK SECURITY	
General security aspects	X.1000–X.1029
Network security	X.1030–X.1049
Security management	X.1050–X.1069
Telebiometrics	X.1080–X.1099
SECURE APPLICATIONS AND SERVICES	
Multicast security	X.1100–X.1109
Home network security	X.1110–X.1119
Mobile security	X.1120–X.1139
Web security	X.1140–X.1149
Security protocols	X.1150–X.1159
Peer-to-peer security	X.1160–X.1169
Networked ID security	X.1170–X.1179
IPTV security	X.1180–X.1199
CYBERSPACE SECURITY	
Cybersecurity	X.1200–X.1229
Countering spam	X.1230–X.1249
Identity management	X.1250–X.1279
SECURE APPLICATIONS AND SERVICES	
Emergency communications	X.1300–X.1309
Ubiquitous sensor network security	X.1310–X.1339
CYBERSECURITY INFORMATION EXCHANGE	
Overview of cybersecurity	X.1500–X.1519
Vulnerability/state exchange	X.1520–X.1539
Event/incident/heuristics exchange	X.1540–X.1549
Exchange of policies	X.1550–X.1559
Heuristics and information request	X.1560–X.1569
Identification and discovery	X.1570–X.1579
Assured exchange	X.1580–X.1589

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

Recommendation ITU-T X.1194

Algorithm selection scheme for service and content protection descrambling

Summary

Recommendation ITU-T X.1194 develops an algorithm selection standard for descrambling in various terminal devices. This Recommendation provides the general service and content protection (SCP) architecture, security requirements and algorithm selection scheme (ASS). In particular, the algorithm selection scheme consists of the SCP control client function, ASS descrambler/demuxer control function and descrambler authentication function.

History

Edition	Recommendation	Approval	Study Group
1.0	ITU-T X.1194	2012-04-13	17

Keywords

Conditional access, descrambling, multiple descrambling, SCP client, service protection, terminal device.

FOREWORD

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

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

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

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

NOTE

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

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

INTELLECTUAL PROPERTY RIGHTS

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

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2012

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

Table of Contents

	Page
1 Scope	1
2 References.....	1
3 Definitions	1
3.1 Terms defined elsewhere	1
3.2 Terms defined in this Recommendation.....	2
4 Abbreviations and acronyms	2
5 Conventions	3
6 Introduction	4
6.1 Internet Protocol television (IPTV) general architecture and content protection architecture	4
6.2 Service protection architecture	5
6.3 General software architecture for terminal devices	6
6.4 Requirement for algorithm selection mechanism for service protection [ITU-T X.1191]	6
7 Algorithm selection scheme	7
7.1 Overview	7
7.2 SCP control client function	10
8 Authentication functions for descrambling algorithm selection.....	19
8.1 ASS descrambler authentication function	19
8.2 Generation and verification procedures for one-time ID	23
9 Control functions for descrambling algorithm selection	24
9.1 ASS descrambler control functions	24
9.2 ASS demuxer control functions.....	28
Annex A – Descrambling algorithm and key delivery message.....	35
A.1 Contents key delivery messages.....	35
A.2 ASS descrambling algorithm.....	37
Appendix I – Descrambling-related standards.....	39
I.1 ATIS IIF default scrambling algorithm (IDSA).....	39
I.2 ETSI TS 103 197 DVB SimulCrypt.....	40
Appendix II – Algorithm selection scheme-related use cases	41
II.1 Definitions of the terms used in the diagram.....	41
II.2 Scenario 1: SCP update	41
II.3 Scenario 2: SCP downloads.....	42
II.4 Scenario 3: Multiple SCP	43
Appendix III – SCP messages in MPEG-2 TS	45
III.1 MPEG-2 TS PSI	45

	Page
Appendix IV – Terminal device provisioning	46
Appendix V – Example of one-time ID generation	48
Bibliography.....	50

.....

Recommendation ITU-T X.1194

Algorithm selection scheme for service and content protection descrambling

1 Scope

Recommendation ITU-T X.1194 develops a set of algorithm selection functions from the existing descrambling algorithms to share terminal devices between service providers and security providers. The scope includes algorithm selection schemes (ASSs) and signalling for selection and interoperability issues and does not include any other schemes of Recommendation ITU-T X.1191.

2 References

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

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

[ITU-T X.1191] Recommendation ITU-T X.1191 (2009), *Functional requirements and architecture for IPTV security aspects*.

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 access control [ITU-T X.800]: The prevention of unauthorized use of a resource, including the prevention of use of a resource in an unauthorized manner.

3.1.2 authentication [ITU-T X.800]: See data origin authentication, and peer-entity authentication.

3.1.3 authorization [ITU-T X.800]: The granting of rights, which includes the granting of access based on access rights.

3.1.4 conditional access [ITU-T X.1191]: The function served by a conditional access system; often used as an abbreviation for conditional access system.

3.1.5 conditional access system [ITU-T X.1191]: A component of a Service and Content Protection system the purpose of which is to prevent unauthorized (unentitled) access to a service or to content.

3.1.6 confidentiality [ITU-T X.800]: The property that information is not made available or disclosed to unauthorized individuals, entities, or processes.

3.1.7 data origin authentication [ITU-T X.800]: The corroboration that the source of data received is as claimed.

3.1.8 digital signature [ITU-T X.800]: Data appended to, or a cryptographic transformation (see cryptography) of a data unit that allows a recipient of the data unit to prove the source and integrity of the data unit and protect against forgery e.g., by the recipient.

3.1.9 integrity [ITU-T X.800]: The property that data has not been altered or destroyed in an unauthorized manner.

3.1.10 key [ITU-T X.800]: A sequence of symbols that controls the operations of encipherment and decipherment.

3.1.11 key management [ITU-T X.800]: The generation, storage, distribution, deletion, archiving and application of keys in accordance with a security policy.

3.1.12 peer-entity authentication [ITU-T X.800]: The corroboration that a peer entity in an association is the one claimed.

3.1.13 scrambling algorithm [ITU-T X.1191]: An algorithm used in a scrambling (encryption) or descrambling (decryption) process.

3.1.14 service and content protection [ITU-T X.1191]: A combination of service protection and content protection, or a system or implementation thereof.

3.1.15 service protection [ITU-T X.1191]: Ensuring that an end user can only acquire a service, and, by extension, the content contained therein, that they are entitled to receive.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 demuxer: Device that takes an input signal and selects one of the output-channels; the selected output channel is connected to the input signal.

3.2.2 descrambler: Device that transposes or decrypts encoded messages or video streams for the terminal or subscribers with privilege.

3.2.3 process: Instance of a computer program that is being executed; contains the program code and its current status.

3.2.4 virtual machine: Software implementation of a machine that can execute programs just like a physical machine; supports separated operating systems.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

A/V	Audio/Video
AAA	Authentication, Authorization, and Accounting
ASS	Algorithm Selection Scheme
CA	Conditional Access
CAS	Conditional Access System
CAT	Conditional Access Table
CBC	Cipher Block Chaining
CW	Control Word
DTCP	Digital Transmission Content Protection
ECM	Entitlement Control Message
EMM	Entitlement Management Message

HMAC	Hash-based Message Authentication Code
IDSA	IIF Default Scrambling Algorithms
IIF	IPTV Interoperability Forum
IPTV	Internet Protocol Television
MK	Master Key
MMS	Multi-Mode Service
MPEG	Moving Picture Experts Group
MPEG-2 TS	MPEG-2 Transport Stream
OTID	One-Time Identifier
PAT	Program Association Table
PES	Packetized Elementary Stream
PGP	Pretty Good Privacy
PID	Program Identifier
PKI	Public Key Infrastructure
PMT	Program Map Table
RAL	Resource Abstraction Layer
SAC	Secure Authenticated Channel
SCP	Service and Content Protection
SIM	Subscriber Identity Module
SP	Service Protection
SSL	Secure Socket Layer
SVM	Secure Virtual Machine
TD	Terminal Device
TK	Terminal's SAC Key
TPS	Triple Play Service
TV	Television
USB	Universal Serial Bus
VM	Virtual Machine
VoD	Video on Demand
VPN	Virtual Private Network

5 Conventions

In this Recommendation:

The keywords "**is required to**" indicate a requirement which must be strictly followed and from which no deviation is permitted if conformance to this document is to be claimed.

The keywords "**is recommended**" indicate a requirement which is recommended but which is not absolutely required. Thus this requirement need not be present to claim conformance.

.....

The keywords "**is prohibited from**" indicate a requirement which must be strictly followed and from which no deviation is permitted if conformance to this document is to be claimed.

The keywords "**can optionally**" indicate an optional requirement which is permissible, without implying any sense of being recommended. This term is not intended to imply that the vendor's implementation must provide the option and the feature can be optionally enabled by the network operator/service provider. Rather, it means the vendor may optionally provide the feature and still claim conformance with the specification.

In the body of this Recommendation and its annexes, the words *shall*, *shall not*, *should*, and *may* sometimes appear, in which case they are to be interpreted, respectively, as *is required to*, *is prohibited from*, *is recommended*, and *can optionally*. The appearance of such phrases or keywords in an appendix or in material explicitly marked as *informative* are to be interpreted as having no normative intent.

6 Introduction

In general, two or more SCPs are deployed on a single terminal device. The content acquired via one SCP system (e.g., from a network) can be accessed via another SCP residing on the same device according to the granted rights. There is a need to support SCP interoperability among multiple security systems that use different security mechanisms to support service and content protection interoperability, thereby maintaining transparency for users.

The object of this Recommendation is to develop a set of functions of algorithm selection schemes from the existing algorithms for content descrambling. This includes the algorithm selection scheme, SCP function, RAL function, interoperability support function and message format.

6.1 Internet Protocol television (IPTV) general architecture and content protection architecture

The general security architecture for IPTV is depicted in Figure 1 below. This general architecture is divided into two primary areas: one considered in-scope for the purpose of considering interoperability based on this Recommendation, and another considered out-of-scope. The first area encompasses the end-user, network provider, and service provider domains, whereas the second area covers the content provider domain.

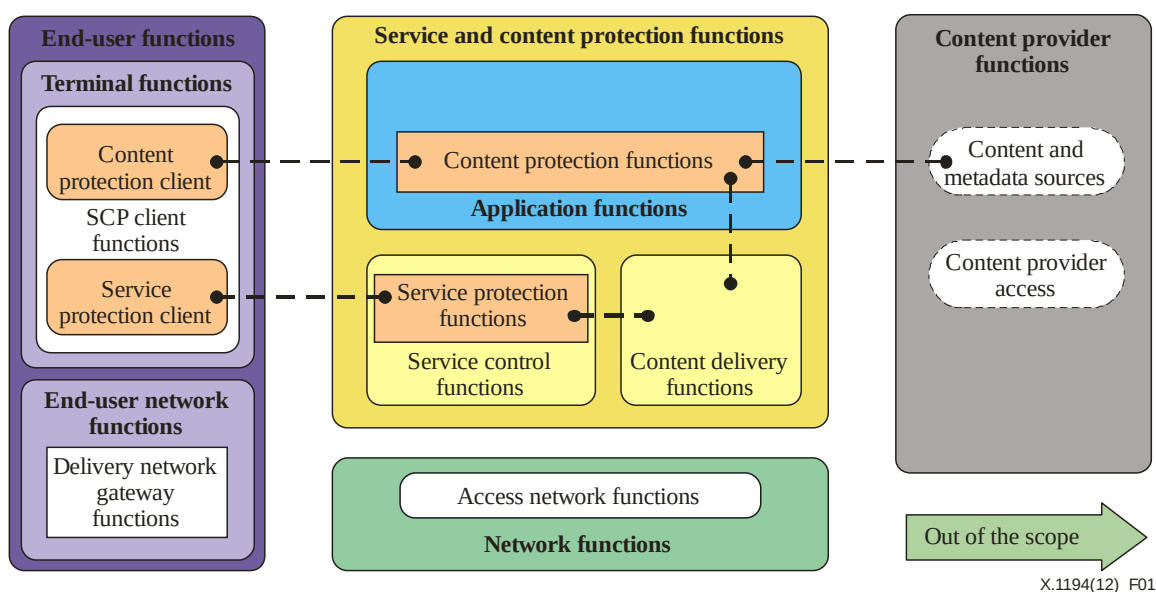


Figure 1 – IPTV general security architecture

A service protection system will in most – if not all – cases, contain mechanisms that can – or do – perform encryption (scrambling) and decryption (descrambling) of both service control signalling traffic and content traffic. Typically, two-way service control traffic will be encrypted in both directions: from server to client and from client to server. On the other hand, content streams will typically be encrypted only from server (service provider) to client (terminal device). Nevertheless, there are usage scenarios wherein a content stream may be uploaded from client to server, in which case such content may be encrypted on a terminal device for uploading purposes (to ensure, for example, that only an authenticated, authorized service provider may access the uploaded content).

6.3 General software architecture for terminal devices

Normally, a terminal device has hardware and software logic to receive and play IPTV contents. The software supports two major functions, e.g., middleware and security. The IPTV middleware supports several functions, including the remote controller, data broadcasting, and media related processing. Another major component is the security function, including the security data session filter and key generation. In the terminal device, the SCP client is only one component of the SCP system.

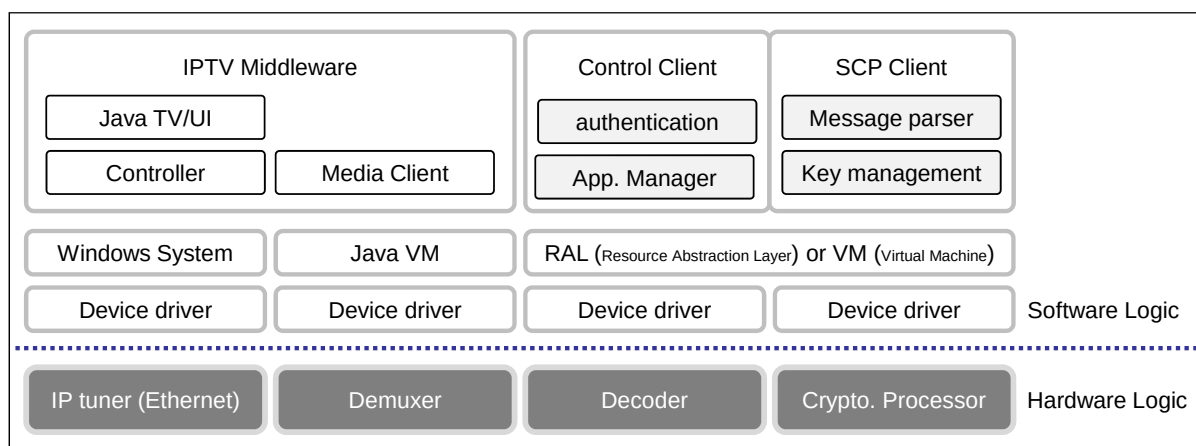


Figure 3 – General software architecture for terminal devices

In Figure 3, the SCP client, one of its security modules, operates based on the resource abstraction layer (RAL) or the virtual machine (VM). A software SCP client can be implemented by native code or application code. Native code means that it does not support portability but is pre-installed on the terminal device. Application code means that the SCP client code runs on a virtual machine.

The Resource abstraction layer (RAL) is an abstraction layer implemented in software between the physical hardware resource and the software that runs on that device. Thus, an SCP client takes unique APIs for hardware access and independent execution. As an isolated duplicate of a real machine, the virtual machine (VM) can provide an instruction set to support the SCP client.

An SCP client can be interoperable among several devices whenever a device's platform is the same. However, in other cases the SCP client is not replaceable.

6.4 Requirement for algorithm selection mechanism for service protection [ITU-T X.1191]

- R 6.1.4-04: The IPTV architecture is required to support mechanisms that allow IPTV services delivery from a third-party provider, as defined in clause 3.
- R 6.3.3-04: The IPTV architecture is required to support the capability to update and query the SCP system concerning scrambling algorithms for IPTV, and any other operator-selected scrambling algorithms.

- R 6.3.3-13: The IPTV architecture is required to support a mechanism to securely retrieve the SCP parameters (e.g., configuration, status) from IPTV terminal device.
- R 6.3.3-14: The IPTV architecture is required to support a mechanism to securely update the SCP parameters (e.g., configuration) of the IPTV terminal device.
- R 6.3.3-16: The IPTV architecture is prohibited from precluding support for the installation and operation of multiple service protection solutions without any hardware replacement except removable devices (such as USB dongle and SIM cards).
- R 6.3.3-19: The IPTV architecture is prohibited from precluding support for a mechanism for the selection of a SCP system from the available SCP systems without any hardware replacement except removable devices.
- R 6.3.3-20: The IPTV architecture is prohibited from precluding support for secure downloading of a SCP system. The specific downloading can optionally depend on specific service protection requirements.
- OR 6.3.2.1-01: Scrambling algorithms for IPTV can optionally apply cryptographic algorithms of different security strength depending on content value.

7 Algorithm selection scheme

The algorithm selection scheme (ASS) provides a common descrambling function with the support of a descrambler control function and a pre-defined ASS descrambling algorithm. In addition, the scrambler provides messages to indicate which algorithm is used. The message depends on the delivery system, such as MPEG-2 TS. In this Recommendation, simple, reasonable message format is provided, and adaptation to other delivery systems is required.

7.1 Overview

Traditional service protection functions are designed with the hardware-based approach. The software-based approach is less secure because the terminal device only has a one-way communication channel. Note, however, that IPTV terminal devices are always online and that they use a two-way communication channel. This means that we can provide a more secure mechanism without hardware support such as security processor.

To remove the hardware dependency, the descrambling function should be separated from the service protection functions. The separated descrambling function is defined by the existing common cryptographic algorithms.

7.1.1 ASS role model for the service protection of linear IPTV

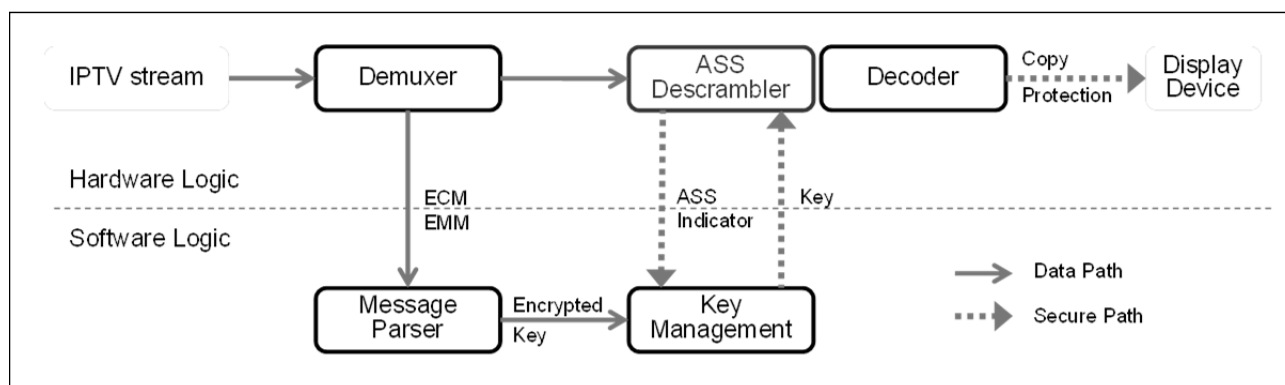


Figure 4 – Key management and ASS descrambler

The relation between an ASS descrambler and a key management function is shown in Figure 4. The key management framework in [b-ITU-T X.1193] generates several keys, and the content decryption key is securely delivered to the ASS descrambler through a secure, authenticated channel.

In Figure 4, the original IPTV stream, which includes linear TV and VoD contents, is delivered to the demuxer (or IP tuner); the demuxer then decodes the received contents and parses the system information defined in the delivery system MPEG-2 TS, because the IPTV stream is encoded with MPEG-TS. In this phase, we extract several stream channels, such as the video stream, audio stream, data stream, and SCP data stream. The information concerning each stream channel is defined in the system information, indicating which packet is which data stream.

After demuxing, the IPTV stream is divided into video/audio stream and data stream; video/audio streams (or only video stream) are scrambled with a key. The ASS descrambler receives a descrambling key from the key management function and descrambles video/audio contents. Decrypted plain video/audio stream is moved to the decoder, which decodes the MPEG contents; the uncompressed video contents are then sent to the display device. Generally, the descrambler and decoder are strongly coupled to minimize the possibility of attack because the data path among these two parties is not protected. This is why those components are implemented with single hardware or single software logic.

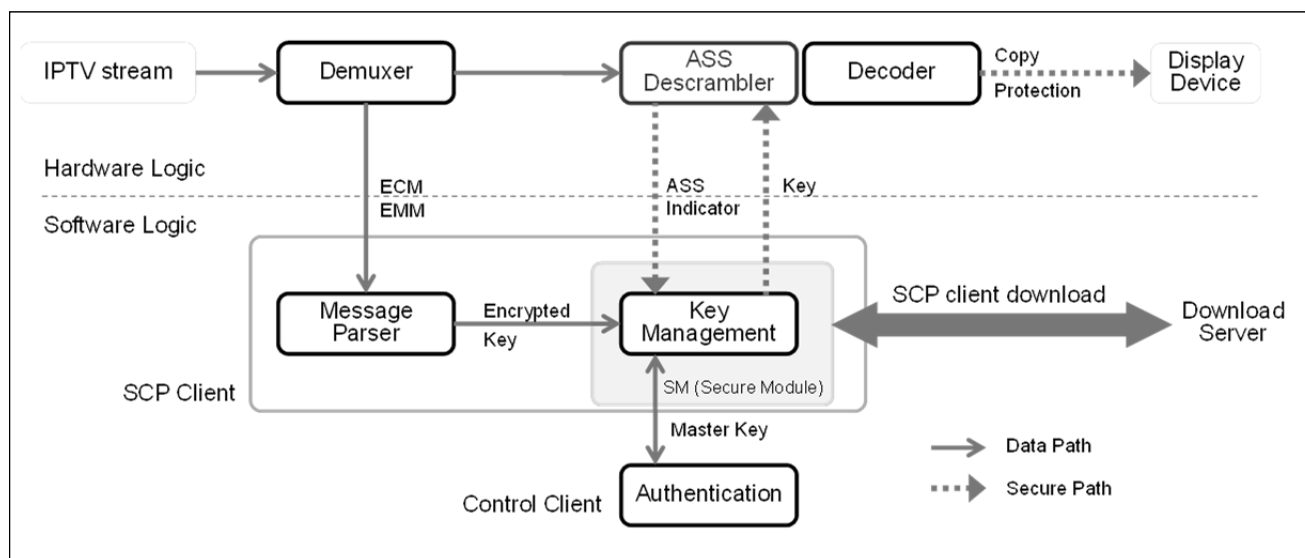


Figure 5 – Downloadable SCP and ASS descrambler

Figure 5 depicts the role of the ASS descrambler in the case of a downloadable SCP. The thick arrow indicates SCP client code download. The thin arrow is content flow, and the dotted arrows are secure communication channels. Each component has the following functions:

- IPTV stream: Integrated contents that include audio, video and data stream.
- Demuxer: This divides the input stream into individual output streams. The audio and video streams are transferred to the ASS descrambler if it is scrambled, and the data stream is sent to the SCP client. Otherwise, the audio and video streams are transferred to the decoder without the descrambling phase.

- SCP client: To decrypt all encrypted streams, the SCP client decodes the data stream and receives SCP-related data, which includes the descrambling algorithm and the decryption key. The SCP client sends those data to the ASS descrambler through a secure communication channel. The SCP client has two functional logics: one is the message parser, which decodes error and key update messages for the ASS descrambler; the other is the secure module (SM), which generates decryption keys within a time period and sends the contents decryption key to an ASS descrambler. The SM provides a key management function, and it is defined in [b-ITU-T X.1193].
- Control client: As a conceptual function for service protection, it includes the client function and the server side function for SCP client management, which involves downloading an SCP client and authenticating the terminal device. After authentication, the SCP client and authentication server can share the master key to make a secure communication channel.
- ASS descrambler: This decrypts all scrambled or encrypted contents. The decryption key is received from the SCP client and the control client. The main role is real-time descrambling and self-protection from several attacks.
- Decoder: This receives the clean audio and video stream. This stream is encoded with a specified algorithm such as MPEG-2 [b-ISO/IEC 13818-1] or ITU-T H.264 [b-ITU-T H.264]. Therefore, the decoder refers to the MPEG media decoder. After decoding, a clean audio and video stream is re-encrypted by a copy protection mechanism such as DTCP between the decoder and display device.
- Display device: Physical display device that supports a copy protection mechanism to receive the protected audio and video data.

Generally, the demuxer, descrambler, and decoder functions are implemented in hardware form, because the system requires real-time processing and no interference from a legacy application.

Figure 6 illustrates the related system components and message flow between components in the terminal device.

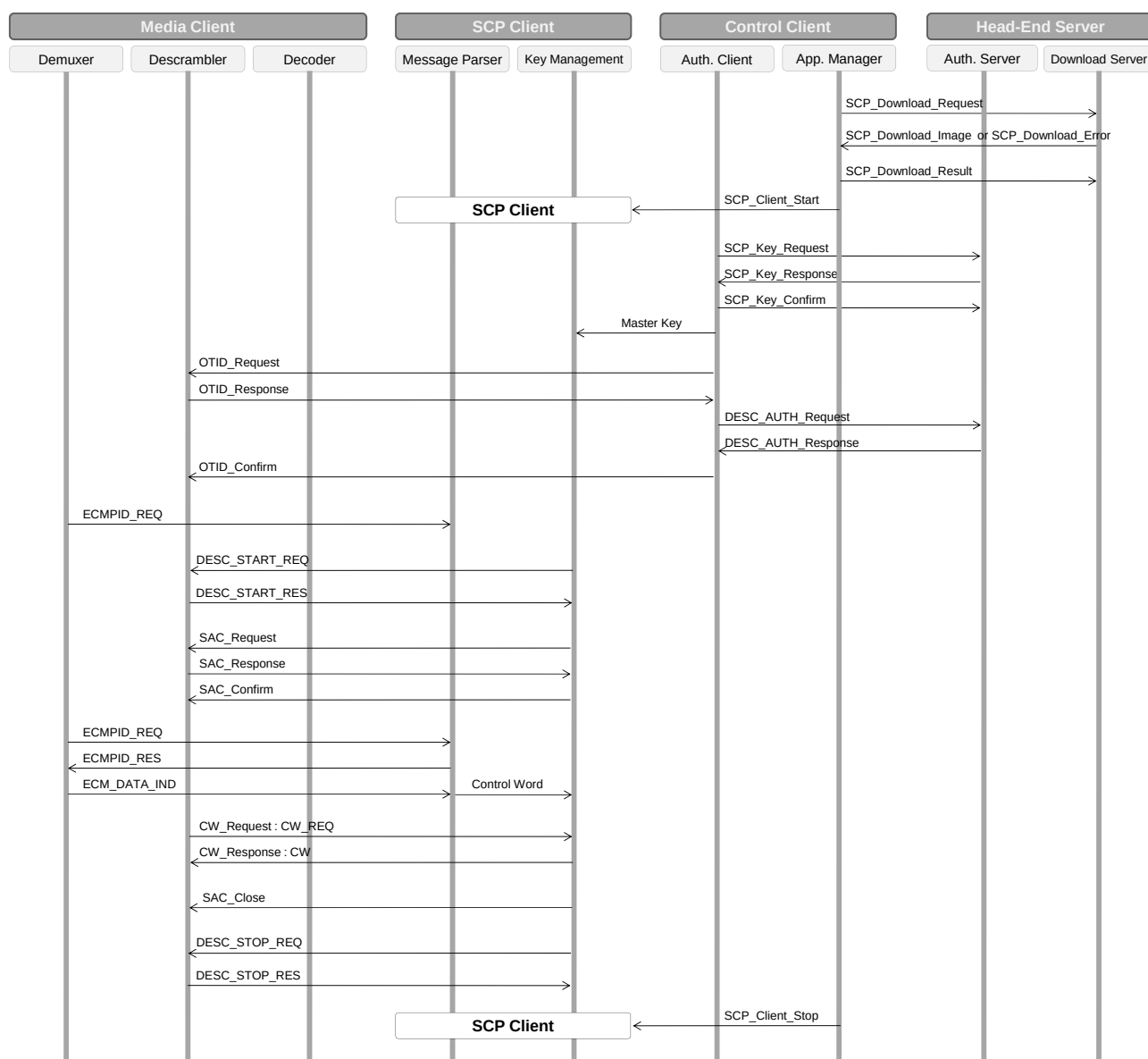


Figure 6 – Message flow of ASS

All messages are divided into three groups: the first group is the SCP control client, which includes SCP downloading and master key distribution; the second group is descrambler control that includes SAC (Secure Authenticated Channel) establishment; the third group is demuxer control.

7.2 SCP control client function

The SCP control client has two functions. One function is the secure downloading and binding of the SCP client to an embedded security module such as a trusted platform, a security kernel, or a hardware security chip. It also controls the life cycle of an SCP client, such as start, pause, stop, and terminate.

The other function is the key distribution protocol. This key is a master key that is used to generate a secure communication channel or authenticate the terminal device as one of the provisioning functions. In the case of the ASS, the main role of the control client is control and delivery of the decryption key to the ASS descrambler.

Therefore, this clause describes some general SCP client functions, SCP client downloading, master key distribution and service protection functions.

7.2.1 Download protocol for the SCP client

In this clause, the procedure for secure SCP client download is defined. SCP clients are developed by the security provider, but delivery is performed by a service provider; hence the need for protection of the SCP client code. A symmetric algorithm is used to ensure confidentiality. HMAC (hash-based message authentication code) is used for integrity.

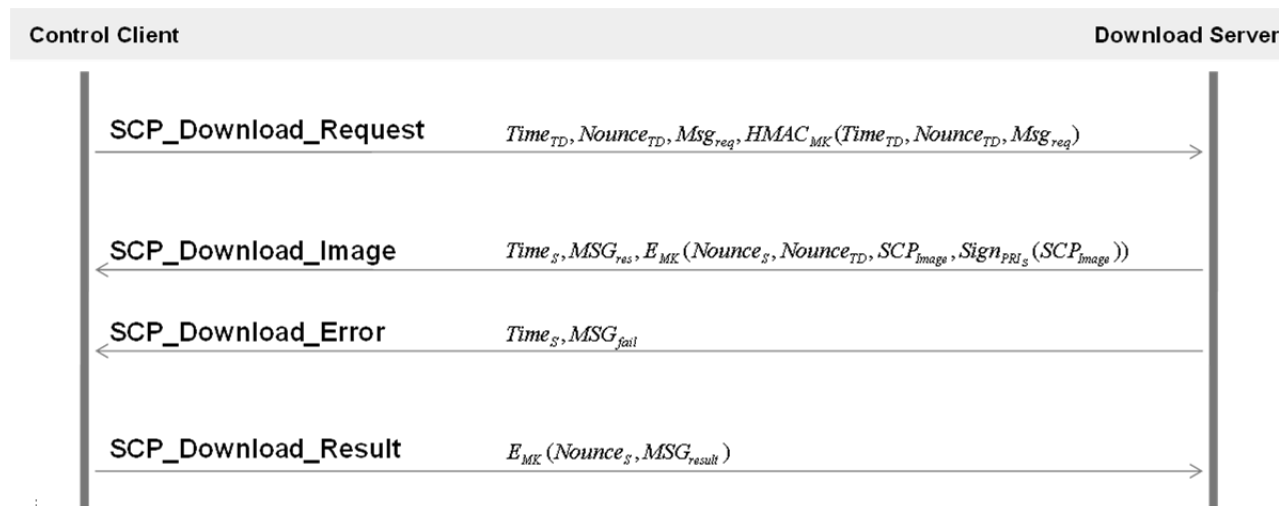


Figure 7 – SCP client download protocol

Figure 7 shows the SCP client download protocol, which consists of four message types. Basically, it has three steps like a 3-way handshake, with an error control step. In this protocol, we use four messages: CA_Download_Request, CA_Download_Image, CA_Download_Error and CA_Download_Result.

EMK(...) is an encryption of AES-128-CBC mode with master key MK. This key is delivered by the key distribution protocol in clause 7.2.2. The details are described in steps P1, P2-1, P2-2, and P3.

P1. SCP_Download_Request: $Time_{TD}, Nounce_{TD}, MSG_{req}, HMAC_{MK}(...)$

- $Time_{STD}$: Terminal device time to avoid a replay attack; it has 14 octets (year/month/day/hour/minute/seconds "YYYY/MM/DD/HH/MM/SS").
- $Nounce_{STD}$: As the random number to avoid a pattern of message data, it is used to generate different signature and encryption bit data with the same input data. To avoid conflict of hash function, we use 40-octet (320 bit) random number.
- MSG_{req} : This is a download request message. The following item, Request_Message, is not fixed for authentication. This message usually delivers information on the terminal device or subscribers such as product model, serial number, and subscriber id. This message has the following items:

{	
Message_type	2*8
Message_length	4*8
Request_Message	Variable
}	

- $HMAC_{MK}(...)$: To guarantee the integrity of download requests, HMAC of a request message is added to the original message. In this message, HMAC is [b-IETF RFC 2104]. Figure 8 shows the details of operation.

Figure 8 – HMAC operation

P2-1. SCP_Download_Image: Time_S , MSG_{res} , E_{MK} , (Nounce_S , $\text{Nounce}_{\text{TD}}$, $\text{SCP}_{\text{Client}}$, $\text{Sign}_{\text{PRIS}}$ ($\text{SCP}_{\text{client}}$))

- *Time_S*: To protect a terminal device from a malicious attacker, server-side time information is added to the message. The time information has 14 octets ("YYYY/MM/DD/HH/MM/SS").
- *MSG_{res}*: The response message of CA_Download_Request has the following items:

```

{
Message_type      2*8
Message_length    4*8
Response_Message  Variable
}

```

- $E_{MK}(Nounce_S, Nounce_{TD}, SCP_{Client}, Sign_{PRIS}(SCP_{client}))$: For secure download, the SCP client (SCP_{client}), service-side nounce $Nounce_S$, terminal nounce $Nounce_{TD}$, and signature of the SCP client are encrypted by the master key (MK). Before the SCP client distribution, the registered SCP client must be verified.

P2-2. SCP_Download_Error: $Time_s$, MSG_{fail}

- *Time_s*: To protect a terminal device from malicious attackers, server-side time information is added to the message. The time information has 14 octets ("YYYY/MM/DD/HH/MM/SS").