CableLabs®

CABLE 3.0

DRM FOR LINEAR CONTENT DELIVERY

prepared by **Dave Belt**
Security Architect
d.belt@cableleabs.com

December 2010

©CableLabs, 2010
All Rights Reserved.
Proprietary and Confidential

DISCLAIMER

This document is published by Cable Television Laboratories, Inc. (“CableLabs®”) to provide information to the cable industry. CableLabs reserves the right to revise this document for any reason including, but not limited to, changes in laws, regulations, or standards promulgated by various agencies; technological advances; or changes in equipment design, manufacturing techniques or operating procedures described or referred to herein. This report is prepared by CableLabs on behalf of its cable operator members to facilitate the rendering, protection, and quality control of communications services provided to subscribers.

CableLabs makes no representation or warranty, express or implied, with respect to the completeness, accuracy or utility of the report or any information or opinion contained in this document. Any use or reliance on the information or opinion is at the risk of the user, and CableLabs shall not be liable for any damage or injury incurred by any person arising out of the completeness, accuracy or utility of any information or opinion contained in this document.

This document is not to be construed to suggest that any manufacturer modify or change any of its products or procedures, nor does this document represent a commitment by CableLabs or any member to purchase any product whether or not it meets the described characteristics. Nothing contained herein shall be construed to confer any license or right to any intellectual property, whether or not the use of any information herein necessarily utilizes such intellectual property.

This document is not to be construed as an endorsement of any product or company or as the adoption or promulgation of any guidelines, standards, or recommendations.

Table of Contents

EXECUTIVE SUMMARY **5**

1 INTRODUCTION **5**

2 PROBLEM SCOPE **5**

2.1 Scope 6
2.2 Out of Scope 6

3 ABBREVIATIONS **7**

4 SYSTEM REQUIREMENTS **7**

4.1 End to End DRM 7
4.2 Container Format 7
4.3 DRM Support 7
4.4 Per Channel Licensing 8
4.5 DVR Licensing 8
4.6 Device Sideloading 8
4.7 Network DVR and Other Linear Network Services 8
4.8 DECE Compatibility 8

5 SYSTEM ARCHITECTURE COMPONENTS **8**

5.1 IP Client Device 9
5.2 Authorization Server 9
5.3 Key Server 9
5.4 Content Server 9
5.5 License Server 10
5.6 Metadata Database 10
5.7 Account Management Server 10

6 KEY AND LICENSE DEFINITIONS **11**

6.1 Device Authentication Certificate (C_{DA}) 11
6.2 Device Encryption Key (K_{DE}) 11
6.3 Service License (L_S) 11
6.4 Service Key (K_S) 11
6.5 Channel License (L_{CH}) 11
6.6 Channel Key (K_{CH}) 12
6.7 Channel Key Package (K_P) 12
6.8 Content License (L_{CN}) 12
6.9 Content Key (K_{CN}) 12

7 CLIENT DEVICE REQUIREMENTS **13**

7.1 Client Software 13
7.2 Unique Device Secrets 13
7.2.1 Device Certificate *13*
7.2.2 Device Encryption Key *13*
7.3 Device Secret Security Requirements 13
7.3.1 Secret Provisioning *13*
7.3.2 Secret Storage *14*
7.3.3 Secret Handling *15*
7.4 Device Account Registration 15

8 CONTENT IDENTIFICATION **15**

8.1 CONTENT IDENTIFICATION 15
8.2 CHANNEL IDENTIFICATION 15

9 USE CASES 16

9.1 DEVICE PROVISIONING 17
9.2 DEVICE REGISTRATION 17
9.3 DEVICE AUTHENTICATION 17
9.4 DEVICE AUTHORIZATION 17
9.5 CHANNEL KEY PACKAGE RETRIEVAL 18
9.6 LINEAR DELIVERY AND CONSUMPTION 18
9.7 LINEAR DELIVERY AND DVR STORAGE 19
9.8 DVR CONTENT PLAYBACK 19
9.9 DVR SIDELOAD 19

List of Figures

FIGURE 1. CONTENT DELIVERY ECOSYSTEM 6
FIGURE 2. LINEAR DELIVERY SYSTEM ARCHITECTURE 9
FIGURE 3. SYSTEM COMPONENT INTERACTION 10
FIGURE 4. KEY PROTECTION NESTING 12
FIGURE 5. LINEAR DELIVERY USE CASE FLOW 16

Executive Summary

An architecture is proposed to allow for end to end delivery of linear video content protected via standard Digital Rights Management for IP based client devices. Included are definitions of headend architectural components, key and license management, client device requirements as well as channel and content identification mechanisms. Use cases supported include linear streaming to an IP based device as well as DVR storage and sideloading to other compliant devices.

1 Introduction

Cable companies are planning linear video content delivery to IP based client devices where the content is protected by an end-to-end DRM system. Architecturally, DRM systems are designed to protect file based or streaming content where there exists a distinct beginning and end to the content, providing a discrete piece of content for licensing and protection. Linear content presents a unique challenge for the application of DRM protections in that the stream is essentially infinite making the assignment of licensing rights and encryption keys a more difficult proposition.

This report proposes an architecture for license and key acquisition of DRM based linear content. All assumptions herein apply not specifically to a particular DRM vendor, but in general to the separate content/license model readily implemented by many DRM vendors.

2 Problem Scope

Figure 1 presents a system level view of the entire content delivery ecosystem. The use cases within this document focus on delivery of linear content to the IP Client Device, as well as sideloading to MSO domain devices.

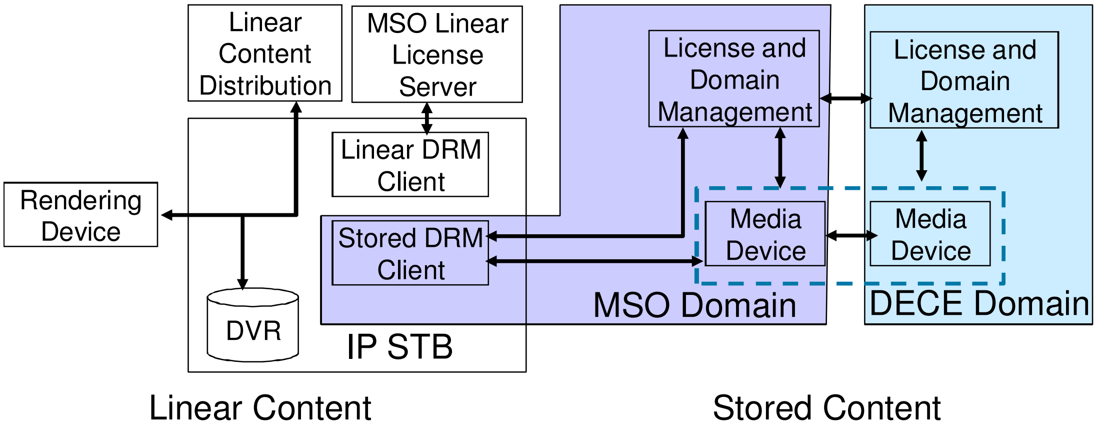

Figure 1. Content Delivery Ecosystem

2.1 Scope

This document focuses primarily on the license acquisition mechanism necessary for the implementation of a linear content distribution system protected by standard DRM technologies. This includes:

- Definition of headend components required for system support.
- Definition of Keys and Licenses required for the secure management of the system.
- High level security requirements for a compatible IP Client Device.
- High level requirements for content and channel identification mechanisms.
- Definition of real time license acquisition for linear content delivery.

2.2 Out of Scope

Various required support mechanisms are discussed herein; however details of their implementation are beyond the scope of this document. Examples of out of scope components include:

- Content Identifiers
- Channel Identifiers
- Metadata Database
- Client Secret Delivery Mechanisms
- Client Secret Protection Mechanisms
- Client Account Registration

Where referenced within the text, these mechanisms will be acknowledged as out of scope.

3 ABBREVIATIONS

CA – Certificate Authority
CRL – Certificate Revocation List
DECE – Digital Entertainment Content Ecosystem
DRM – Digital Rights Management
DVR – Digital Video Recorder
EIDR – Entertainment Identifier Registry
EPG – Electronic Program Guide
IP – Internet Protocol
MSO – Multiple Systems Operator
OTP – One Time Programmable
PC – Personal Computer
PKI – Public Key Infrastructure
STB – Set Top Box
TLS – Transport Layer Security
VSP – Video Service Package

4 SYSTEM REQUIREMENTS

The following sections outline a set of high level system requirements used as a basis for the architecture presented herein. These requirements were compiled by CableLabs via active discussion with its member companies.

4.1 END TO END DRM

Content is protected at the Cable headend via a DRM system. The content protection mechanism remains intact through distribution and consumption by the end user.

4.2 CONTAINER FORMAT

The DRM protection mechanism shall work with Fragmented File over IP as well as MPEG2 TS container formats.

4.3 DRM SUPPORT

The proposed architecture shall specifically support usage of the following DRM systems.

1) Microsoft PlayReady
2) Adobe Flash Access
3) Apple Fairplay

More generally, the architecture shall also support any DRM endorsed by the DECE ecosystem in support of Section 4.8.

4.4 Per Channel Licensing

License rights for linear content shall be assigned on a per channel basis.

4.5 DVR Licensing

All content delivered via linear channel shall inherit a common DVR right, defining if the content may be stored and played back on the receiving device only.

4.6 Device Sideloading

The linear license delivery mechanism shall support content sideloading from the receiving device. Licensing rights to do so may be outside of the scope of the linear channel itself, and may require additional license acquisition to perform the sideload.

4.7 Network DVR and Other Linear Network Services

The linear license does not apply to network DVR, start over, go back or other services provided by the network as, from a client perspective, these are on-demand services. User rights to these types of network services can be controlled by applications and/or on-demand content rights which are out of scope of this report.

4.8 DECE Compatibility

The linear delivery architecture shall be compatible with the DECE ecosystem.

5 System Architecture Components

Figure 2 provides a system level view of the DRM-based linear delivery system. Each component identified herein consists of a logical unit, of which multiple units may comprise a single physical server within the system. All sub-sections within Section 5 refer back to this diagram.

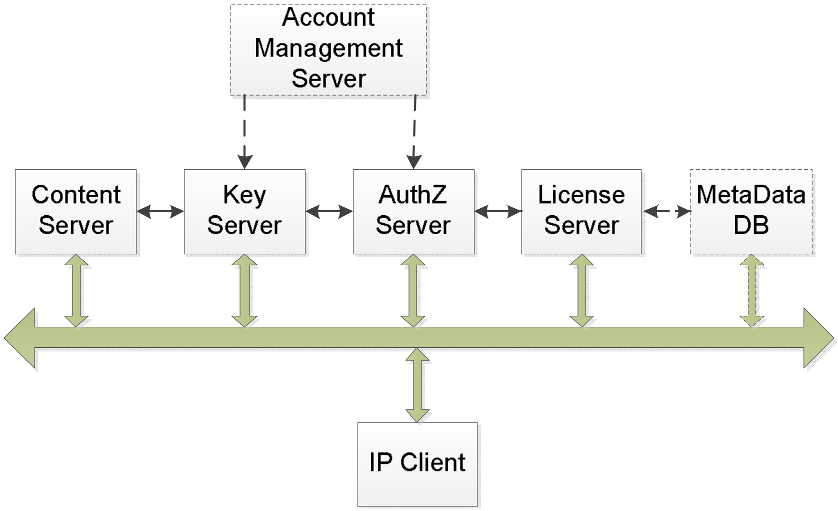

Figure 2. Linear Delivery System Architecture

5.1 IP Client Device

The IP Client Device serves as the content consumer of the system. At minimum, the device requires IP connectivity as well as onboard client software for decode and display of linear content. The device could take the form of an STB, PC, television, or portable device. Additional requirements for the IP Client are covered in Section 7.

5.2 Authorization Server

The Authorization Server provides a registered IP Client Device with the credentials necessary to view the content available under the Customer's Video Service Package (VSP). This is achieved by issuing Service Licenses and Keys to the device as defined in Sections 6.3 and 6.4.

5.3 Key Server

The Key Server issues Channel Key Packages to the IP Client Device as defined in Section 6.7. The device identifies its service via the Service License and receives the Channel Key Package in return, providing the comprehensive set of keys required for decryption of the channels within the VSP.

5.4 Content Server

The Content Server streams live linear video content to the IP Client Device. Each channel stream within the VSP is encrypted utilizing its appropriate Channel Key, defined in Section 6.6, and obtained from the device's Channel Key Package.

5.5 License Server

The License Server issues specific licenses to the IP Client Device for content elements identified within the linear stream. This license facilitates sideloading to a secondary portable device.

5.6 Metadata Database

The Metadata Database provides content and channel metadata to the IP Client Device based on unique Content and Channel ID's as defined in 8.1 and 8.2. The metadata delivered is then used to render an EPG on the device and provide the customer the ability to select channels for playback. The Metadata Database is discussed herein for the purpose of providing a comprehensive picture of the system architecture; however its actual implementation is beyond the scope of this effort.

5.7 Account Management Server

The Account Management Server provides information on the customer's VSP and maintains accounting of devices registered with the account. The Account Management Server is discussed herein with regards to registration of devices with the account, as well as device authentication, however, its actual implementation is beyond the scope of this effort.

Figure 3 provides the top level interactions of the system level components as they relate to the use cases outlined in Section 9.

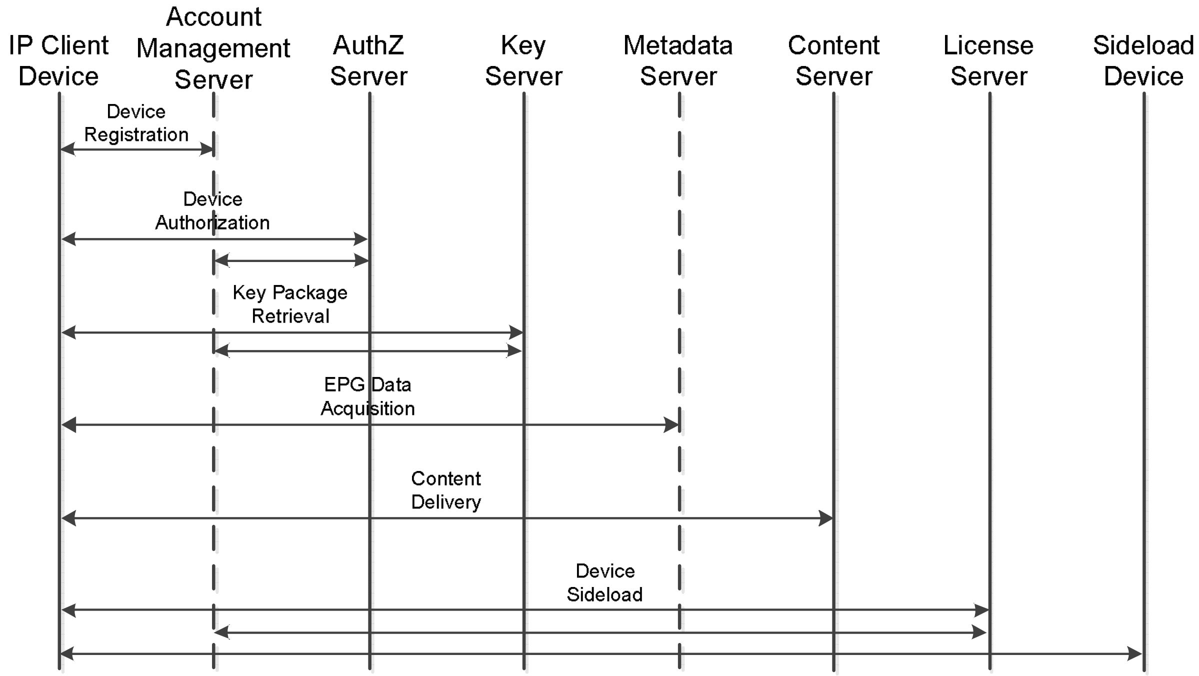

Figure 3. System Component Interaction

6 Key and License Definitions

The following sections identify the keys and licenses utilized by the architecture.

6.1 Device Authentication Certificate (C_{DA})

A unique per device certificate used to authenticate the device with the headend. The certificate shall be signed via PKI Certificate Authority to permit device authentication with the system headend. The Device Authentication Certificate shall have the lifetime of the device or the client application supporting linear content playback. Being derived via a trusted PKI, the certificate may be revoked via CRL.

6.2 Device Encryption Key (K_{DE})

A unique per device public/private key pair used to encrypt the Service Key (K_S) during delivery from the headend. The private key shall be cryptographically protected, per Section 7.3.2, during storage and handling on the client device at all times in order to prevent compromise. The Device Encryption Key (K_{DE}) shall have the lifetime of the device or the client application supporting linear content playback.

6.3 Service License (L_S)

A unique licensing certificate identifying the channels to which the consumer is subscribed. The Service License (L_S) shall be signed via PKI Certificate Authority in order to verify its authenticity. The L_S shall have a medium length lifetime, i.e., 7 - 30 days, to be implementation-specific and correspond to the expiration of the Service Key (K_S).

6.4 Service Key (K_S)

A unique symmetric key used to protect the delivery of content keys for a particular VSP. The Service Key (KS) shall be cryptographically protected, using the Device Encryption Key KDE, during storage and handling on the client device at all times in order to prevent compromise. The KS shall have a medium length lifetime and be determined by the Service License (LS).

6.5 Channel License (L_{CH})

A licensing certificate conveying the rights for a single linear channel within the VSP. An individual Channel License (L_{CH}) shall utilize the Channel Identifier defined in Section 8.2 for identification and contain a single unique Channel Key (K_{CH}). The L_{CH} shall be delivered as part of a larger K_{CH} Package and have a short length lifetime, i.e., 15 - 120 mins.

6.6 Channel Key (K_{CH})

A unique per channel symmetric key used to protect the content delivered within a single channel. The Channel Key (K_{CH}) shall be protected using the Service Key (K_S) during transport and storage on the client device. Delivery is part of the Channel License (L_{CH}) with its lifetime defined therein.

6.7 Channel Key Package (K_P)

A package of licenses, keys and channel identifiers as defined in Section 8.2 defining all rights associated with the Consumer's VSP. The Channel Key Package (K_P) shall have a short length lifetime as dictated by the expiration times of the individual channel licenses contained therein.

6.8 Content License (L_{CN})

A licensing certificate conveying the rights for a single piece of content delivered within a linear channel. An individual Content License (L_{CN}) shall contain a single unique Content Key (K_{CN}) and be retrieved from the headend at the time of side load to another device. The Channel License (L_{CH}) shall have a lifetime as dictated by the MSO & Content Provider license agreements.

6.9 Content Key (K_{CN})

A unique symmetric key used to protect an individual piece of content delivered within a single channel. The Content Key (K_{CN}) delivery is part of the Content License (L_{CN}) with its lifetime defined therein.

Figure 4 presents an overview of the key nesting, utilizing the Device key to protect the Service key which in turn protects the Channel key.

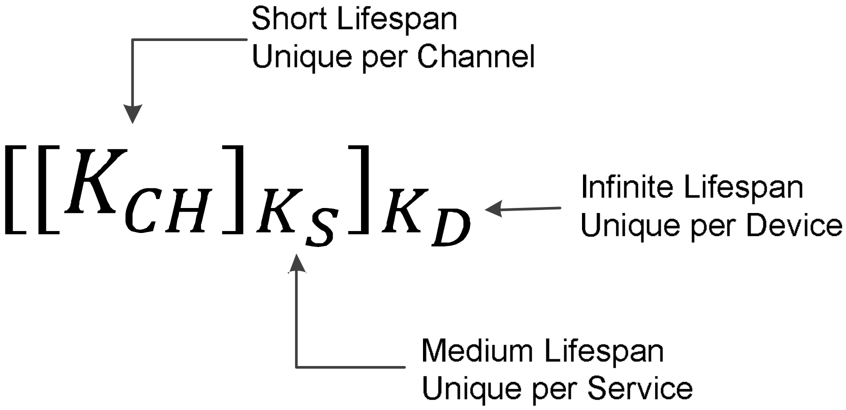

Figure 4. Key Protection Nesting

7 Client Device Requirements

The IP Client Device, serving as the main content consumer of the system, has a number of requirements that must be met in order to function with the linear content delivery system.

7.1 Client Software

The client software for playback of linear video content must support the use cases defined within Section 9. Additionally, the client software must provision, handle and protect all keys discussed herein in a cryptographically secure manner as recommended in Section 7.3. Due to the wide variety of devices with the potential to serve as clients, the actual client software architecture is beyond the scope of this effort.

7.2 Unique Device Secrets

7.2.1 Device Certificate

Each IP Client Device shall employ a unique signed Device Authentication Certificate (C_{DA}), as defined in Section 6.1 and issued via a PKI hierarchy. C_{DA} shall be the utilized to register the device to a particular customer account as well as to authenticate the device as described in Section 9.3.

7.2.2 Device Encryption Key

Each IP Client device shall employ a unique Device Encryption Key (K_{DE}), as defined in Section 6.2. K_{DE} shall be used in the delivery of the Service Key (K_S) as described in Section 9.4.

7.3 Device Secret Security Requirements

The IP Client Device is required to handle and store various service and content keys as defined in Section 6. The following provides guidelines for the handling and storage of the secrets discussed herein within the IP Client Device.

7.3.1 Secret Provisioning

The Device Authentication Certificate (C_{DA}) and Encryption Key (K_{DE}) defined in Section 7.2.1 and Section 7.2.2 are specified to be unique per device. The secrets are to be issued via a credible CA and provisioned into the device via cryptographically secure manner. Due to the wide range of device capabilities, the actual method of provisioning is beyond the scope of this effort, however some preferred method are outlined as follows.

7.3.1.1 Factory Provisioning

Assuming that the device will be a known consumer of DRM based linear content; the ideal method of provisioning of the device secrets is at manufacture. Via a factory floor key server, device secrets are injected into the device and stored in a cryptographically secure manner prior to leaving the factory.

7.3.1.2 Secure Download with Existing Authentication Mechanism

Assuming Section 7.3.1.1 is not a feasible provisioning method, device secrets may be delivered to the device via a secure link assuming the device has some known secret such as an OTP key or existing PKI certificate. The IP Client Device establishes a secure TLS session with a provisioning server and authenticates itself utilizing its existing secret. The device secrets of Section 7.2 are then downloaded to the device and stored in a cryptographically secure manner.

7.3.1.3 Secure Download with No Existing Authentication Mechanism

The least secure method of provisioning is on an IP Client Device with no unique secret and should be used as a last resort. In this case, the IP Client Device establishes a secure TLS session with a provisioning server and prompts the customer via the Client Application for a username and password associated with the customer account. Upon authentication, the device secrets of Section 7.2 are then downloaded to the device and stored in a cryptographically secure manner.

The provisioning methods outlined are listed in order of secure preference. It is up to the integrator based on their own organizational security requirements which level is appropriate and/or permissible.

7.3.2 SECRET STORAGE

Storage of secrets on the IP Client Device is highly device specific; however, some preferred guidelines are provided as follows.

7.3.2.1 Hardware Protection

Many modern microprocessors employ security processors utilizing protected storage as well as encryption engines. If protected storage is available, all device secrets shall be stored within this space. If a hardware based encryption engine is available, preferably tied to a unique per device key, all device secrets shall be encrypted with the hardware engine. If both hardware mechanisms are available, they both shall be used.

7.3.2.2 Software Protection

Assuming no hardware protection is available, the device must protect the device secrets via software. If a unique per device key is available on the system, it shall be used to protect the device secrets. In the absence of a unique key, one shall be generated utilizing a unique identifier, e.g., processor ID, as a seed. The unique identifier shall be input into a secret hash and the result used to encrypt the device secrets. Extreme care must be exercised in the execution and handling of the hash function via proper vaulting in the development process as well as software obfuscation on the client device itself. Additionally any code encrypting or decrypting the device secrets shall be obfuscated to prevent unauthorized disassembly.

7.3.3 Secret Handling

At any time that device secrets are handled in the clear by software, the code performing the handling shall be obfuscated to prevent unauthorized disassembly.

7.4 Device Account Registration

The IP Client Device shall provide a mechanism to register the device with the customer's VSP account. The actual mechanism for doing so is implementation dependent and out of the scope of this effort; however, the Device Authentication Certificate C_{DA} shall be associated with the account at the headend to enable proper device authentication as defined in Section 9.3.

8 Content Identification

8.1 Content Identification

All linear content shall be uniquely identified via a content registry system. The preferred method for this identification is through the EIDR content identification system; however, other systems such as DECE or proprietary mechanisms may also be used.

8.2 Channel Identification

All linear channels shall be uniquely identified via a network registry system. The preferred method for this identification is through the EIDR Network Identifier; however other proprietary mechanisms may also be used.

9 USE CASES

The following outlines the use cases required for DRM-based linear content delivery. Figure 5 provides a general use case flow to which all sub-sections refer.

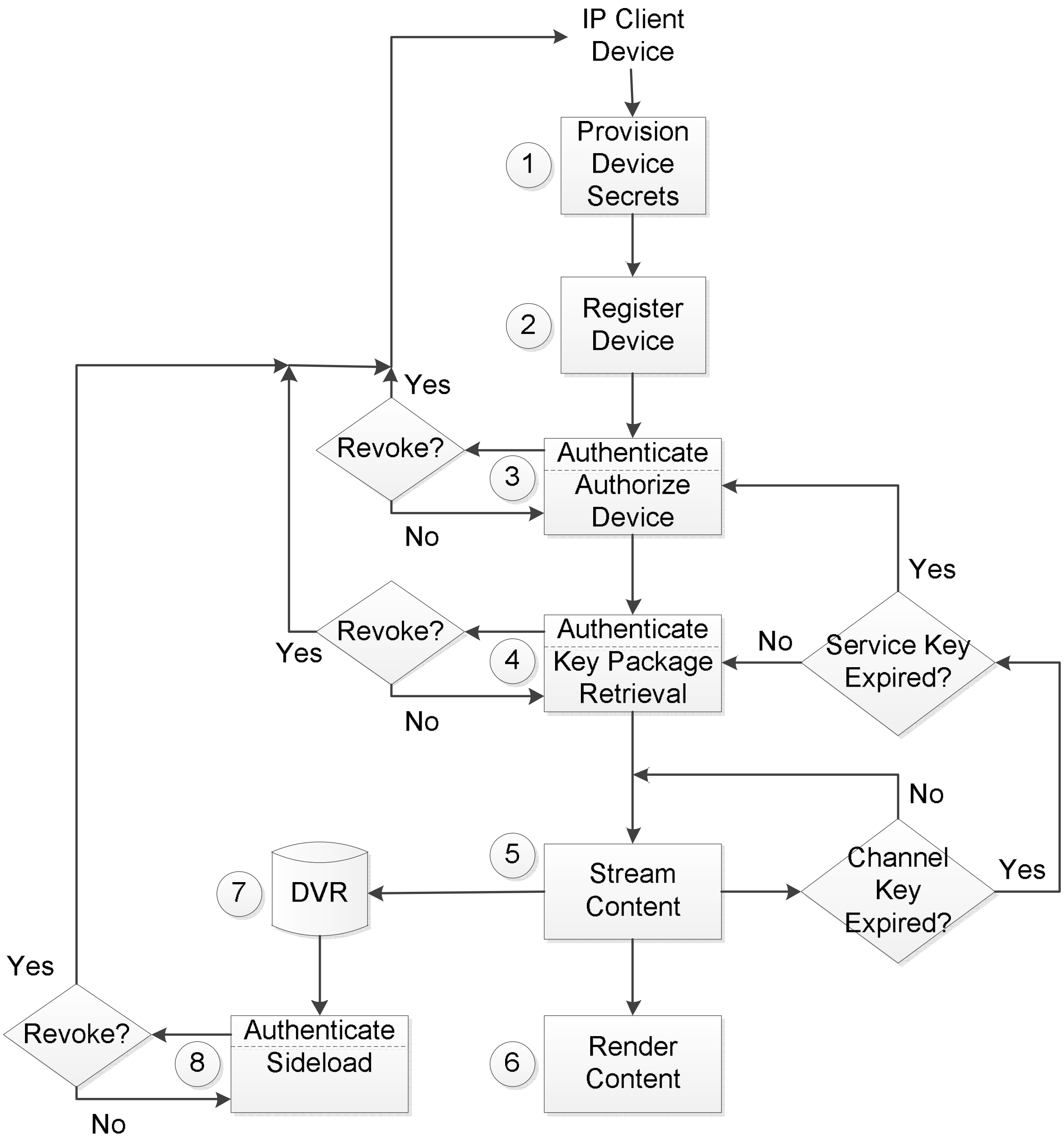

Figure 5. Linear Delivery Use Case Flow

9.1 Device Provisioning

In Figure 5, (1) identifies the initial step of device secret provisioning. As defined in Sections 6.1 and 6.2, each client device requires a Device Authentication Certificate (C_{DA}) and Device Encryption key (K_{DE}) in order to interoperate with the DRM based linear delivery system. These secrets must be injected into and stored within the device in a cryptographically secure manner and are highly device architecture dependent. As such, actual secret provisioning is beyond the scope of this effort; however, it should follow the guidelines as outlined in Section 7.3.

9.2 Device Registration

In Figure 5, (2) identifies the process of device registration with the customer account. The actual details of the registration process are beyond the scope of this document; however, it shall be required that a cryptographic hash of the Device Authentication Certificate (C_{DA}) shall be registered with the account which will be used thereafter for device authentication.

9.3 Device Authentication

Device authentication shall occur when the device connects to the authorization (3), key (4) and license servers (8) as shown in Figure 5. The authentication procedure shall be as follows.

1) Connect to server.
2) Transmit the Device Authentication Certificate (C_{DA}) to the server for authentication.
3) Validate the C_{DA} signature. On failure, disconnect.
4) Check C_{DA} against the CRL. On match, disconnect. Additionally, on revocation, the IP Client device should either mark C_{DA} as invalid, or delete it completely, effectively forcing a disable or re-provision.
5) Validate the C_{DA} hash against that stored in the customer account as registered in Section 9.2. On failure, disconnect.
6) On successful authentication, proceed with expected server functionality.

9.4 Device Authorization

Once registered, the device obtains its authorization credential, shown in (3) of Figure 5, as follows.

1) Connect to Authorization Server.
2) Authenticate the device per Section 9.3.
3) Transmit the Device Public Encryption Key (K_{DE}) to the Authorization Server.
4) Retrieve the Service License (L_S) and Service Key (K_S) corresponding to the customer's VSP.
5) Asymmetrically encrypt K_S with K_{DE}.
6) Return L_S along with the encrypted K_S to the IP Client Device.
7) Store L_S and K_S as delivered. There are no special requirements for secure storage of K_S as long as it remains encrypted using K_{DE}.

As defined in Section 6.3 and Section 6.4, the Service License (L_S) and Service Key (K_S) have a medium length lifetime. Upon expiration, the client is required to renew both by repeating the process described.

9.5 Channel Key Package Retrieval

Once authorized, the device retrieves the current Channel Key Package, shown in (4) of Figure 5, associated with the customer's VSP as follows.

1) Connect to the Key Server.
2) Authenticate the device per Section 9.3.
3) Transmit the Service License (L_S) to the Key Server.
4) Retrieve the current Key Package (K_P) associated with the customer's VSP and identified by the (L_S).
5) Return K_P to the IP Client device.
6) Store K_P as delivered. There are no special requirements for secure storage of K_P as long as the keys remains encrypted using Service Key (K_S).

As defined in Section 6.5 and Section 6.6, the Channel Licenses and Keys have a short length lifetime. Upon expiration, the client is required to renew both by repeating the process described.

9.6 Linear Delivery and Consumption

Once the Key Package is obtained, the customer may now view video content as shown in (5) and (6) of Figure 5. It is assumed that the IP Client Device renders some manner of EPG support via Channel and Content IDs outlined in Section 8, however it is beyond the scope of this effort. The Customer selects a channel for playback wherein delivery and consumption proceed as follows.

1) The IP Client Device decrypts the Service Key (K_S) using the Private Device Encryption Key (K_{DE}).
2) The IP Client Device retrieves the encrypted Channel Key (K_{CH}) from the Key Package (K_P).
3) The IP Client decrypts K_{CH} using the symmetric K_S.
4) The IP Client connects to the Content Server and initiates streaming of the selected channel.
5) The Content Server streams the encrypted channel content to the IP Client Device.
6) The IP Client Device decrypts the content using K_{CH}.
7) The IP Client Device renders the decrypted content via its associated display device.
8) Prior to expiration of the channel package, the Client Device must obtain a new Key Package (K_P) per Section 9.5.

It should be noted that steps 1), 3), and 6) are considered cryptographically sensitive operations. As such, all code performing these operations is subject to the measures outlined in Section 7.3.3.

9.7 LINEAR DELIVERY AND DVR STORAGE

As an alternative, or in conjunction with Section 9.6, the customer may wish to record content to a DVR storage mechanism, as shown in steps (5) and (7) of Figure 5. Using the same assumptions as in Section 9.6, the DVR recording proceeds as follows.

1) The IP Client connects to the Content Server and initiates streaming of the selected channel.
2) The Content Server streams the encrypted channel content to the IP Client Device.
3) The streamed content is written to storage as downloaded from the Content Server.
4) The Content Identifier, delivered within the stream, is stored with the content to be used later for content identification purposes.
5) The Channel Key (K_{CH}) is stored with the content encrypted with the symmetric Service Key (K_S).
6) K_S is stored with the content, encrypted with the Public Device Encryption Key (K_{DE}).

9.8 DVR CONTENT PLAYBACK

Once content is recorded to the DVR storage, playback of the content occurs as follows.

1) The IP Client Device retrieves the Service Key (K_S) stored with the content on the DVR.
2) The IP Client Device decrypts K_S using the Private Device Encryption Key (K_{DE}).
3) The IP Client retrieves the Channel Key (K_{CH}) stored with the content on the DVR.
4) The IP Client Device decrypts K_{CH} using $K_{S.}$
5) The IP Client Device streams the content from the DVR storage, decrypting it using $K_{CH.}$
6) The IP Client Device renders the decrypted content via its associated display device.

It should be noted that steps 2), 4), and 5) are considered cryptographically sensitive operations. As such, all code performing these operations is subject to the measures outlined in Section 7.3.3.

9.9 DVR SIDELOAD

Content recorded to the DVR storage may be sideloaded to other compatible portable devices via peripheral interface, as shown in step (8) of Figure 5. The Sink Device must meet the same client requirements as the Source Device, as outlined in Section 7. Given a compatible device, the sideload proceeds as follows.

1) The Sink Device connects to the Source Device via peripheral interface.
2) The Customer selects content for transfer via either device's UI, the details of which are beyond the scope of this effort.
3) The Sink Device transmits its Device Authentication Certificate (K_{DA}) and Public Device Encryption Key (K_{DE}) to the Source Device.
4) The Source Device connects to the License Server.
5) The Source Device authenticates itself per Section 9.3.
6) Using the Sink Device's K_{DA}, the Source Device authenticates the Sink Device to the License Server per Section 9.3.

7) The Source Device retrieves the Content ID for the requested content from the DVR storage and transfers it to the License Server along with the Sink Device's $K_{DE.}$
8) The License Server issues a portable device Content License (L_{CN}) for the requested content.
9) The License Server encrypts the Content Key (K_{CN}) with the Sink Device's K_{DE}, inserts it into the L_{CN} and signs the $L_{CN.}$
10) The License Server transfers L_{CN} to the Source Device.
11) The Source Device transfers the encrypted content from DVR storage along with L_{CN} to the Sink device.
12) The Sink Device stores the content and L_{CN} within DVR storage for later playback.