Adobe® Access™

Version 4.0

April 2012

Overview

© 2012 Adobe Systems Incorporated. All rights reserved.

This guide is protected under copyright law, furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated. Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in the informational content contained in this guide.

This guide is licensed for use under the terms of the Creative Commons Attribution Non-Commercial 3.0 License. This License allows users to copy, distribute, and transmit the user guide for noncommercial purposes only so long as (1) proper attribution to Adobe is given as the owner of the user guide; and (2) any reuse or distribution of the user guide contains a notice that use of the user guide is governed by these terms. The best way to provide notice is to include the following link. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/3.0/

Adobe, the Adobe logo, Adobe AIR, Adobe Access, Flash Player, and Flex are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Apple and Mac OS are trademarks of Apple Inc., registered in the United States and other countries. Java is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and other countries. Linux is the registered trademark of Linus Torvalds in the U.S. and other countries. Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. All other trademarks are the property of their respective owners.

Updated Information/Additional Third Party Code Information available at http://www.adobe.com/go/thirdparty.

Portions include software under the following terms:

This product contains either BSAFE and/or TIPEM software by RSA Security Inc.

Adobe Systems Incorporated, 345 Park Avenue, San Jose, California 95110, USA.

Notice to U.S. Government End Users. The Software and Documentation are "Commercial Items," as that term is defined at 48 C.F.R. §2.101, consisting of "Commercial Computer Software" and "Commercial Computer Software Documentation," as such terms are used in 48 C.F.R. §12.212 or 48 C.F.R. §227.7202, as applicable. Consistent with 48 C.F.R. §12.212 or 48 C.F.R. §§227.7202-1 through 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (a) only as Commercial Items and (b) with only those rights as are granted to all other end users pursuant to the terms and conditions herein. Unpublished-rights reserved under the copyright laws of the United States. Adobe Systems Incorporated, 345 Park Avenue, San Jose, CA 95110-2704, USA. For U.S. Government End Users, Adobe agrees to comply with all applicable equal opportunity laws including, if appropriate, the provisions of Executive Order 11246, as amended, Section 402 of the Vietnam Era Veterans Readjustment Assistance Act of 1974 (38 USC 4212), and Section 503 of the Rehabilitation Act of 1973, as amended, and the regulations at 41 CFR Parts 60-1 through 60-60, 60-250, and 60-741. The affirmative action clause and regulations contained in the preceding sentence shall be incorporated by reference.

Last updated 6/20/2012

Contents

An overview of Adobe Adobe Access SDK

About Adobe Access 1
Terminology and core concepts 2
Content distribution workflow 3
Key features 6
Adobe Access components 7
Deploying Adobe Access 9

An overview of Adobe Adobe Access SDK

Adobe® Access™ SDK is a digital rights management platform that makes it possible to protect and securely deliver video and audio content for playback on consumer devices such as personal computers. Adobe Access is a flexible platform that enables content owners to protect their content and maintain control over distribution. Content owners can protect and manage their rights by creating licenses for each digital media file, ensuring that a wide variety of the highest-quality content is made available to consumers.

Adobe Access supports a wide range of business models, including video on demand, rental, and electronic sell-through. You can distribute content protected with Adobe Access by streaming through Adobe Flash Media Server software, offering progressive download via HTTP using Adobe's HTTP Dynamic Streaming technology, or permitting downloads to a content library for local playback at the consumer's convenience.

Topics in this document are:

"About Adobe Access" on page 1

"Terminology and core concepts" on page 2

"Content preparation" on page 3

"Content acquisition" on page 4

"Key features" on page 6

"Adobe Access components" on page 7

"Deploying Adobe Access" on page 9

About Adobe Access

Adobe Access lets content owners and distributors control how and where their content can be distributed and experienced, providing end-to-end protection throughout the content lifecycle. It encrypts Flash FLV and F4V video files, which can then be streamed or downloaded to Microsoft® Windows®, Apple® Mac OS®, Linux®, and Android® platforms, and enforces business models such as online video rental. Consumers can enjoy high-quality content at their convenience, whether they are online or offline. Adobe Access offers key advantages that content owners and distributors can use to quickly deliver new services that differentiate them from the competition.

This cross-platform solution can be integrated into existing billing and authentication systems, allowing content owners and distributors to maintain control of audio and video content no matter how widely it is distributed. Adobe Access helps protect and monetize content with confidence and provides the flexibility to manage how and where protected content is experienced.

Adobe Access gives:

- Content owners and distributors broader reach for media assets, and a way to recuperate production costs while protecting those assets from misuse.
- Content distributors the ability to rapidly distribute content and recuperate production costs through direct (user-paid) or indirect (advertiser-paid) compensation.
- Consumers the flexibility to enjoy premium content with a rich viewing experience that is intuitive, non-intrusive, convenient, and engaging.

- Developers the ability to automate Adobe Access capabilities, including authorization and authentication, into a content distributor's existing environment.

Terminology and core concepts

The following terms and concepts are used throughout this document:

Consumer

The *consumer* is the end user who downloads or streams content.

Content

Content consists of digital audio or video files encoded in the FLV or F4V format.

Content Encryption Key

The *Content Encryption Key* (CEK) is a cryptographic key used to encrypt the content.

Content owners

Content owners are the business entities who own the copyright to the content. These can be large motion picture studios, or smaller, independent producers of movies or other audiovisual content.

Content packagers

Content packagers are organizations who package content for use with Adobe Access. Content owners or Distributors may choose to package their own content, or they may enlist the services of a third-party to package their content, and distribute it electronically via the Internet.

Digital certificate

Digital certificates (also referred to as *certificates*) bind an entity, such as an individual, organization, or system, to a specific public and private key pair. Digital certificates can be thought of as electronic credentials that verify the identity of an individual, system, or organization.

Digital signature

A *digital signature* binds the publisher's identity to the content that they have published and provides a mechanism to detect tampering. Digital signature algorithms use cryptographic hash functions and asymmetric—or public/private key pair—encryption algorithms. Some digital signatures also take advantage of digital certificates and public key infrastructure (PKI) to bind public keys to the identities of content owners or distributors.

Distributor

Distributors (also referred to as *content distributors* or *retailers*) are business entities who secure distribution rights from content owners to publish and circulate content to consumers. In some cases, the same entity is both the content owner and content distributor.

DRM metadata

Information that the client, meaning Adobe® Flash® Player or Adobe® AIR® runtime, sends to identify the requested content.

License

A *license* is a data structure that contains an encrypted key used to decrypt content associated with a policy. The license is generated by Adobe Access when the consumer requests content, and is bound to the consumer's computer. Using a policy as a reference, the license defines the rights available to the consumer who downloads content. In order to view content, the consumer must obtain a license.

License acquisition

License acquisition is the process of acquiring a license allowing the consumer to decrypt and view protected content according to a set of usage rules. License acquisition occurs when a client sends information identifying the requested content (the *DRM metadata*) and the machine certificate (identifying the consumer's computer) to the License Server (see below).

License Server

The *License Server* may be integrated into the distributor's or service provider's billing and authentication systems, and may contain business logic to verify that the consumer requesting protected content is authorized to view the content. If the user is authorized to access the content, the License Server issues a license allowing the runtime client to decrypt and playback content based on the policy and rights associated with the consumer's account.

You must create and deploy a License Server using the Adobe Access SDK.

Policy

A *policy* is a container for the usage rules that determine how consumers can use protected content. Policies are defined independently of the content being protected. A policy does not enforce rights until it is bound to the content through the license. A policy lists the set of usage rules, meaning the permissions or "rights" that consumers have to the content they acquire. For example, content owners may create a policy that ensures that protected content is only accessible by consumers for a specific period of time. This policy is then applied to all the content for which the content owner wants to enforce this restriction.

Policies are created using Adobe Access SDK.

Protected content

Protected content (also referred to as *packaged content*) refers to FLV and F4V video content that has been encrypted using Adobe Access SDK or other supported tools.

Retailers

See the entry for *distributors* earlier in this section.

Content distribution workflow

Any use of Adobe Access consists of two key steps in different points of the workflow. *Content preparation* must be done once per asset, and results in creating protected content. *Content acquisition* is done multiple times, once for every consumer that wants to watch that protected asset.

Content preparation

Before making content available for distribution, you must first encode the content in the FLV or F4V video format, create one or more policies specifying usage rules for the content, and package the content using Adobe Access SDK.

The steps to encode, package, and distribute content are as follows:

1 Encode the content in the FLV or F4V format using encoding tools available from Adobe or from third parties.

2 Create policies specifying the usage rules under which consumers can view the content.

   A *policy* is the container for the rules and restrictions that determine how, when, and where protected content can be viewed by consumers.

   The packager requires at least one policy with at least one usage rule. You can override the usage rule, and add additional usage rules, when the License Server generates the license.

3 Package the content and specify what policies to apply.

   Adobe Access SDK encrypts the content using a Content Encryption Key (CEK), and binds one or more policies to the content. The result is a *protected content file* that can only be played by a consumer who has obtained a license from the corresponding License Server.

   During packaging, the content is encrypted using the CEK. The CEK is encrypted using the License Server public key and included in the DRM metadata along with the policies. The DRM metadata is signed using the Packager private key, and the metadata is included in the protected content.

4 Make the protected content available for distribution to consumers.

   The protected content is typically distributed using a content distribution network (CDN). The CDN can use any mechanism supported by the client runtime, such as Flash Media Server, Adobe HTTP Dynamic Streaming for multiple bitrate streaming, or an HTTP Web Server for progressive download.

Content acquisition

When a consumer acquires a protected content file from a website or CDN, the consumer must also acquire a license that contains a key to decrypt the video before it can be played. The following steps illustrate a common workflow for how protected content is accessed by a computer running Flash Player or Adobe AIR:

1 The consumer visits the retailer's website, and selects a video to watch. The consumer attempts to download or stream the protected video to their computer using either Flash Player or an Adobe AIR application.

   If this is the first time the consumer has attempted to access protected content using this specific computer, the Flash Player or Adobe AIR runtime must first be individualized as described in Step 2. If the runtime client has already been individualized, the process of acquiring a license occurs as described in Step 3.

2 The Flash Player or Adobe AIR runtime client acquires a unique digital certificate (called a *machine certificate*) from an Adobe-hosted server.

   This process of assigning a unique certificate is called *individualization*. Individualization uniquely identifies both the computer and the Flash Player or Adobe AIR runtime used to playback content.

   The individualization process allows the downloaded licenses to be bound to a specific computer on which the client is installed. Every computer is given a unique machine credential (machine private key and machine certificate). If a specific client were to become compromised, it can be revoked and barred from acquiring licenses for new content.

3 The client parses the protected content as it begins to download or stream to the consumer's computer, and extracts the URL of the retailer's License Server from the DRM metadata embedded within the file. The DRM metadata is typically separate from the content, such as embedded in an accompanying manifest file or as a binary blob, but can also be embedded within the content file. The client contacts the License Server at the specified URL, and acquires a license (as described below in Step 4).

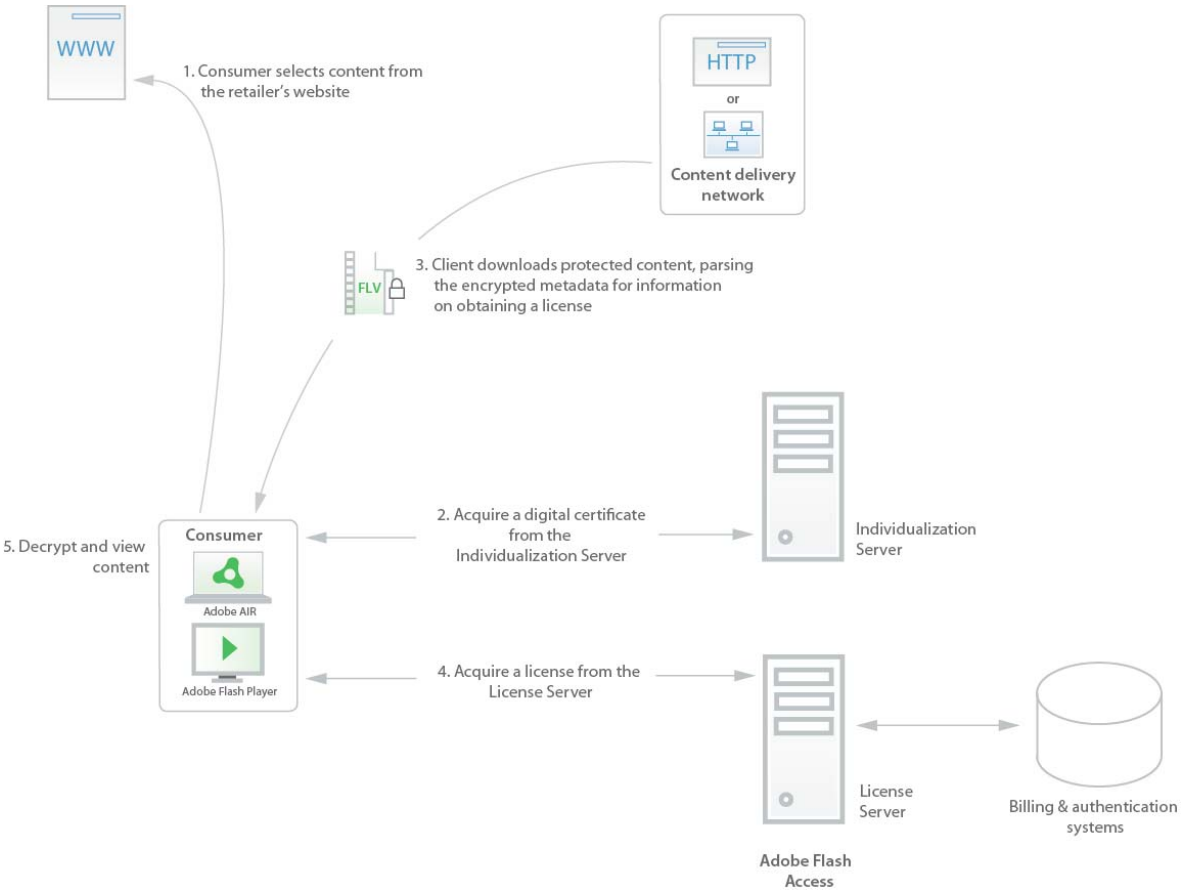
4 The client acquires a license from the retailer's License Server.

During license acquisition, the client sends information identifying the requested content (the *DRM metadata*) and the machine certificate (identifying the consumer's computer) to the retailer's License Server. The license request sent to the server is encrypted using the Transport public key, which is also included in the DRM metadata.

The License Server — which may be integrated into the retailer's billing and authentication infrastructure — can perform a business rules check to verify that the user is authorized to view the requested content. If the business rules allow it, the License Server issues a license containing the content encryption key to decrypt the content and the usage rules associated with that user's account. To process a license request, the License Server decrypts the request using its Transport private key. The CEK in the metadata is decrypted using the License Server private key, and re-encrypted to bind the license to the device making the request. The license is signed using the License server private key. The license response is signed using the Transport private key, and encrypted before being returned to the client.

If allowed by the license, the client stores the license to enable *offline access* to the license. License caching allows the consumer to view protected content without reacquiring a license every time they want to view content.

5 Once the Flash Player or Adobe AIR runtime client has a license, the client extracts the CEK from the license, and the consumer can view the content they are authorized to access.



The previous example shows just one possible workflow. Alternatively, you might use a workflow with a proactive download of content where the license acquisition happens much later. Another option is to implement a pre-order workflow where the license acquisition occurs before the content is accessed.

Key features

Adobe Access provides these key features:

Remote iOS key delivery: Support for delivering remote iOS keys through a multi-tenant key server, called the Adobe Access Key Server.

Persistent content protection: Content remains protected throughout the distribution chain. Once the content is packaged, it remains protected at all times, and portions of it are only decrypted at the time of playback and in accordance with the usage rules.

Because the content is packaged with usage rules and licensing information, protection always travels with the content. If unlicensed consumers attempt to play the content, the policy embedded in the content redirects them so they can acquire a valid license for the content.

Secure playback of protected content: Proven encryption schemes are used to protect content from unauthorized playback.

Flexible usage rules: Usage rules determine how consumers can use protected content. The usage rules supported by Adobe Access allow for several different business models, including pay-per-view, movie rental, subscriptions, or ad-funded services. The usage rules are specified by the policy you embed into the content during packaging, or can be specified by the License Server during license acquisition.

Output protection: Output protection controls can be used to engage protection schemes such as HDCP, CGMS-A or Rovi (formerly Macrovision) ACP. This can help protect content output over digital (for example, HDMI, DVI, and UDI) and analog (for example, S-Video, and Component Video) outputs.

You can specify output protection options in the policy you create for your content, allowing or disallowing access to content based on a device's output protection capabilities. For example, you can specify that output protection is not required for certain content, but require output protection for premium video content, preventing the output of content on operating systems that lack this capability.

While these rules are consistently enforced across all platforms, currently it is only possible to securely turn on output protection on Windows platforms.

Support for dynamic streaming: Use the dynamic streaming feature of Flash Media Server or Adobe HTTP Dynamic Streaming to protect content encoded at multiple bit rates. Dynamic streaming uses a video stream that dynamically adjusts the bit rate and playback quality based on available bandwidth, providing an improved user experience. Adobe Access makes it possible use the same Content Encryption Key and license across the different encodings of the same asset.

Offline viewing: The Adobe AIR runtime client allows consumers to download and store content for later viewing, whether or not the computer remains connected to the Internet.

Identity-based licensing: Links permissions to user identities, requiring consumers to authenticate themselves (providing a user id and password) in order to gain access to content. Identity-based licensing requires that the party developing the client application implement user authentication as part of the license acquisition workflow.

License chaining: Allows consumers to extend the life of all of their content by renewing a single root license. One of the primary uses of license chaining is for subscription-based business models, where, at the end of a subscription period, licenses to large numbers of content files can be renewed by simply renewing the root license.

Both root and leaf licenses are bound to the consumer's computer. The leaf license contains the CEK and a reference to the root license. The root license can extend the rights granted in the leaf license. For example, a consumer may have leaf licenses for 50 pieces of content that will expire at the end of a given subscription period. If the consumer downloads a new root license with a new expiration date, all 50 pieces of content can be played back until the updated license expires.

Adobe Access 2.0 introduced license chaining in which both the leaf and root licenses are bound to a specific machine. Adobe Access 3.0 introduces an enhanced form of license chaining, in which a leaf is bound to a root license, and only the root license is bound to a specific machine or domain. Read *Enhanced License Chaining* in *Using the Adobe Access Server for Protecting Content*.

Key rotation: During typical packaging, the content is encrypted using the Content Encryption Key (CEK), and the client obtains a license containing the CEK in order to consume the content. When key rotation is enabled, the key used to encrypt the content can be changed so that the key is only used to encrypt a portion of the content. Read *Key Rotation* in *Using the Adobe Access Server for Protecting Content*.

Out-of-band Licenses: With Adobe Access Professional, it is possible to implement a workflow in which clients obtain pre-generated licenses out-of-band, eliminating the need to deploy a License Server.

Domain Support: As an alternative to binding a license to a specific device, Adobe Access supports binding licenses to a domain. Multiple devices may join a domain and receive a domain token allowing licenses to be moved between devices in the domain. Read *Domain Registration* in *Using the Adobe Access Server for Protecting Content*.

Partial encryption: Specifies whether all frames, or only a subset of frames, should be encrypted. There are three levels of encryption: low, medium and high. Reducing the encryption level may decrease CPU overhead on lower end machines.

Disconnected content packaging: Content packaging does not require a network connection to the License Server. This enables secure back-end operations by limiting the exposure of compressed content that has not yet been protected.

Control over clock rollback: Adobe Access provides for the secure and accurate calculation of time to detect clock rollback on the client computer. This enforces rights related to accessing content for a specific amount of time, and prevents consumers from subverting access rights by altering the time on their computer.

Individualization: Allows binding content to a particular machine.

Application whitelist: Allows the client runtime to ensure that protected content only plays within an authorized SWF or AIR application.

Revocation of compromised clients: Compromised client software can be revoked. If the Flash Player or Adobe AIR runtime is determined to have been compromised, service can be refused to those clients until they upgrade to a newer and more secure version of the client software.

Multiple policies for the same video file: A single piece of video content can have multiple policies embedded during packaging. When issuing a license, the License Server may decide which of the policies to use, enabling a content distributor to use the same protected file for different business models (such as rental and electronic sell-through).

Adobe Access components

The main components of Adobe Access 3.0 include a Java SDK and the Flash Player and Adobe AIR client runtime environments.

In Adobe Access 3.0, two forms of the Java SDK are available:

- Adobe Access Core SDK
- Adobe Access Professional SDK

For more information on setting up the SDK, see Setting up the SDK in *Using the Adobe Access Server for Protecting Content.*

The Adobe Access SDK lets you develop a digital rights management solution that integrates with your organization's existing business infrastructure, such as content management, billing, and user access control systems. Flash Player and Adobe AIR let you create and easily deploy applications through which consumers can access and view large libraries of digital content.

Adobe Access SDK

Adobe Access is delivered as a Java SDK that provides the building blocks from which you can create a server implementation. Using the SDK you can create an Adobe Access solution suited to your organization's business model.

The Java APIs provided in the SDK are:

Java APIs for managing domains

These APIs are used to allow the server to handle client requests for joining and leaving domains.

Java APIs for protecting content

These APIs are used to define rights and prepare content for distribution. The content protection APIs are:

- Policy management

  The policy management API is used to create and modify policies to be applied to content. Policies can be created or updated, including getting/setting all usage rules and allowing additional parameters in a custom namespace.

- Content packaging

  The content packaging API is used to encrypt content and retrieve metadata from the packaged content.

Java APIs for issuing licenses

These APIs are used when a client requests a license from the server. The SDK supports the following requests from the client:

- Authentication

  The authentication API can be used to handle authentication requests and generate authentication tokens.

- License generation and acquisition

  The license generation and acquisition API is used to generate a license for the user.

- Support for Adobe AIR version 1.5 clients and content

  For the purposes of backwards compatibility, the SDK has APIs to handle requests from AIR applications created for use with AIR version 1.5 and earlier clients and protected content.

Reference implementation

The SDK includes a reference implementation, a simple Adobe Access deployment that demonstrates how to use the Java APIs. The reference implementation provides a License Server, Watched Folder Packager, Adobe Access Manager AIR application, and command line tools for content packaging and policy management based on the Java APIs. To learn more about the Adobe Access reference implementation, see *Protecting Content*.

Adobe Access Server for Protected Streaming

For streaming use cases where content is protected with Adobe Access, such as for Adobe HTTP Dynamic Streaming, the software also includes Adobe Access Server for Protected Streaming. This solution can be easily deployed on a servlet container such as Tomcat and can achieve a high level of scalability and performance to meet the largest content distribution needs.

Adobe Flash Player

Flash Player is a lightweight browser plug-in and runtime that delivers consistent and engaging user experiences, stunning audio/video playback, and pervasive reach. Flash Player provides high-quality playback of streamed or downloaded video content. For content publishers, Flash Player provides the means to customize the playback screens surrounding content, allowing deeper branding experiences and monetization through advertising using banners and overlays. For consumers, Flash Player presents an intuitive and visually appealing way to view video content.

For more information on Flash Player, visit: www.adobe.com/go/flashplayer

Adobe AIR

Adobe AIR is a cross-operating system runtime that allows content producers to extend their existing investments in the web to the desktop by designing customized multimedia applications. Built on proven, open technologies, it provides a reliable, simplified way for businesses to develop and deploy custom applications that can be trusted to deliver a more secure, enjoyable user experience. Adobe AIR allows businesses to easily integrate rich media to create a more immersive and interactive user experience. It lets developers use familiar tools such as HTML, JavaScript, Flash, or Adobe® Flex® software to deploy their unique combination of rich Internet applications to either Windows, Macintosh, or Linux. Businesses have complete control of the user interface, and they can design a user experience to reflect and reinforce their brand. With built-in support for playback of content protected with Adobe Access SDK, Adobe AIR helps create custom, end-to-end content distribution chains.

For more information on Adobe AIR, visit: www.adobe.com/go/air

Deploying Adobe Access

A key advantage to Adobe Access SDK is that you can install it on any Java™ application server or servlet container, such as Tomcat. You also need JDK™ 1.5 or higher. For more information on software requirements, see Adobe Access SDK platform requirements.

The high-level steps to deploy Adobe Access are:

1. Install and configure Adobe Access SDK.
2. Obtain digital certificates from Adobe.
3. Create a License Server using the SDK, or deploy Adobe Access Server for Protected Streaming.

4 Create content packaging and policy management tools to package content, use the provided content preparation tools, or license one of the Adobe HTTP Dynamic Streaming packagers.

5 Define usage rules for your content, and create policies in support of those rules.

6 Package your content using the packaging and policy management tools.

7 Develop video applications with which consumers can view your protected content using Flash Player or Adobe AIR, or use an established OVP (Online Video Platform) that supports Adobe Access.

8 Deploy a SWF file for use with Flash Player to your website, or post the Adobe AIR installer for download.

These steps are expanded upon in the following sections, with references to other documents containing additional information.

Deploying on a 64-bit operating system

A 64-bit operating system, such as the 64-bit version of RedHat or Windows, provides much better performance than a 32-bit operating system.

Install Adobe Access SDK

Adobe Access SDK is provided as a Java archive (JAR) file. To learn more about installing Adobe Access, see *Using Adobe Access Server for Protecting Content* and *Secure Deployment Guidelines*.

Implement a License Server

Using Adobe Access SDK, you must create a License Server. When content is protected using Adobe Access, it cannot be viewed until a license is issued to the consumer by the License Server. If identity-based licensing is used, password-based authentication ensures that only authorized consumers can open and view content.

When implementing a License Server, you must obtain the necessary digital certificates from Adobe. Refer to the Adobe Access Certificate Enrollment document for detailed instructions on requesting certificates.

To learn more about implementing a License Server, and obtaining digital certificates, see *Using Adobe Access Server for Protecting Content.*

Create content packaging and policy management tools

Using the Adobe Access SDK, you can create content packaging and policy management tools. The policy management APIs allow administrators to create, view details of, and update policies. The packaging APIs embed the policy into the FLV or F4V file, and encrypt the file using the content encryption key.

The Adobe Access SDK includes a reference implementation that provides examples of content packaging and policy management tools.

To learn more about creating content packaging and policy management tools, see *Protecting Content*.

Create policies and package content

Using the usage rules supported by the SDK, you must define and create policies in support of your organization's business model, and then package your content using those policies. Once policies are applied to content during packaging, you can maintain control of your content no matter how widely it is distributed.

The policies in Adobe Access support a wide range of different usage rules, including:

- Specifying the number of days a license is valid once a consumer begins watching the content.

- Allowing license caching.
- Specifying client runtimes and versions that can access content (for example, users must have a certain Adobe AIR application or a specific version of Flash Player).
- Requiring that consumers authenticate themselves using a username and password prior to viewing protected content, or allowing anonymous access.

To learn more about packaging content, see *Protecting Content*. To learn more about the usage rules and the business models they support, see *Usage Rules*.

Develop applications for video playback

To enable consumers to access and view content, develop a video playback application using Flash Player or Adobe AIR. Once you developed a video playback application, you must deploy it to consumers. If you are developing an application using Flash Player, host it on your organization's website. If you are developing an application using Adobe® AIR®, post the AIR application installer so that consumers can download and install the application on their computer.

To learn more about developing custom video playback applications for use with Adobe Access, see the "Working with Video" chapter in ActionScript 3.0 Developer Guide, the Adobe Video Technology Center, and the Open Source Media Framework.

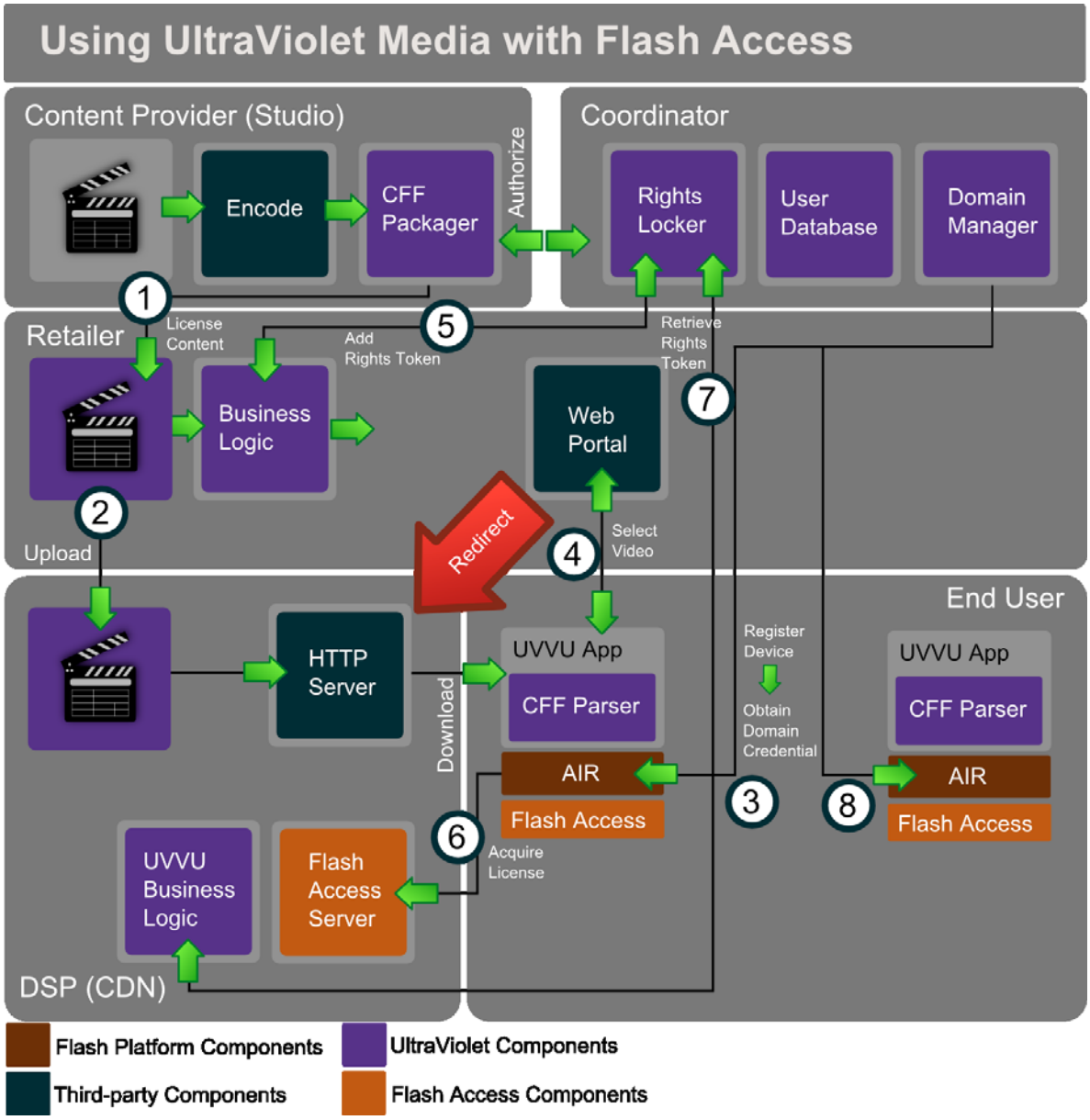
Additional Deployment Scenarios

Adobe Access can be used with other third-party content streaming solutions to set up a complete and secure DRM-based media distribution ecosystem.

UltraViolet media and Adobe Access

UltraViolet is a digital rights authentication and cloud-based distribution system that can enable consumers of digital home entertainment content to stream and download purchased content through multiple platforms and devices. UltraViolet content will be downloaded (or streamed) in a Common File Format, (CFF) using Common Encryption (CENC).

It is easy to set up an UltraViolet system along with Adobe Access. The following use case depicts the content flow behavior:



1 The content owner encodes and packages the content in CFF. The packaged content is licensed to a retailer for distribution.

2 The retailer uploads the content to a digital service provider, like CDN. The content is now available for download. Note that some of these roles can be played by one or more companies.

The end user has a device that supports Adobe AIR. In addition to that, the user needs to install an UltraViolet-compliant application. The application includes the necessary code to parse the CFF and present it for consumption by the runtime. All the sensitive cryptographic operations are handled in the secure runtime.

3 The application can trigger a domain join for the device, which interacts with the co-ordinator. The co-ordinator maintains a rights locker, a user database and domains. The co-ordinator's domain manager is built using the Adobe Access SDK to implement Adobe Access-specific domain join/leave operations.

4 The user can then use the application to select a video they want to acquire from the retailer. The retailer typically provides a web portal and handles all business logic.

5 The retailer then interacts with the co-ordinator to add a rights token. The retailer then redirects the request to the service provider for the actual content download.

6 If the device does not yet have a license to the content, it triggers a license request using the CFF. The request typically includes a domain certificate, user credentials and information about the application. The service provider operates a Adobe Access License Server (developed using the Adobe Access SDK) that follows the UltraViolet specifications.

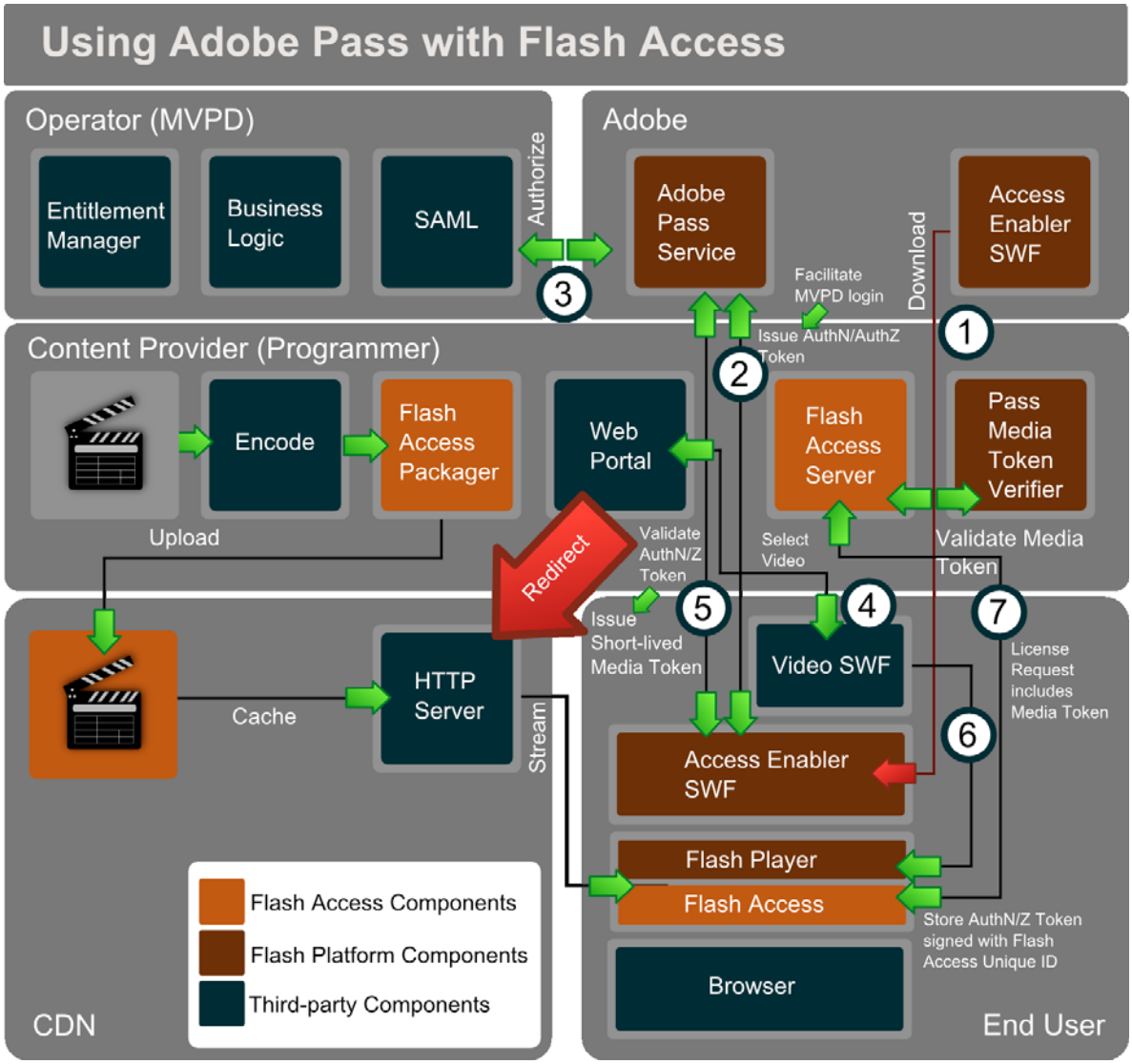
7 The service provider's UltraViolet business logic interacts with the co-ordinator as needed to retrieve the appropriate rights token to determine whether a content license should be issued.

   The content license is bound to the domain. The client application can insert the license into the CFF file. Content can now be played back in the application, with all protection and usage rule enforcement handled by the Adobe Access component in the runtime.

8 Other devices and applications owned by the same end user can be registered with the co-ordinator. The content can now be loaded in other Adobe Access devices without requiring any external transaction.

Adobe Pass and Adobe Access

Adobe Pass provides user/device authentication and authorization across multiple content providers. The user must have a valid cable TV or satellite TV subscription.



Adobe Pass can be used along with Adobe Access for protecting the media content. In this scenario, The video player (SWF) can load another SWF called the *Access Enabler*, which is hosted by Adobe Systems. The *Access Enabler* is used to connect to the Adobe Pass service, and facilitate SAML SSO integration with MVPD's (Multichannel Video Programming Distributor) identity provider systems. This involves redirecting the user's browser briefly to the MVPD login page, then persisting an AuthN token and finally return to the content web site with a cached AuthN session.

The *Access Enabler* can then facilitate backend authorizations between Adobe Pass service and the MVPD. The MVPD maintains the business logic and determines what content the user is entitled to. The entitlement is persisted in an additional AuthZ token for that content resource and is sent back to the client.

The authentication and authorization tokens are signed using the unique ID and private key of the Adobe Access client to avoid tampering or spoofing. This token can only be accessed via the *Access Enabler*.

The video player can trigger the process by calling `getAuthorization` on the *Access Enabler*. When valid AuthN/AuthZ tokens are present, the *AccessEnabler* issues a callback to the video player that will include a short-lived media token for playing the video content.

Adobe Pass provides a media token validator Java library that can be deployed to a server. When using the Flass Access server for content protection, you can integrate the media token validator with a Adobe Access server-side plug-in to automatically issue a generic license after successfully validating the media token. The content is then streamed from the CDN servers to the client. To acquire a content license, the short-lived media token can be submitted to the Adobe Access server, where the validity of the token is verified and a license can be issued.

The long-lived AuthN token is used generally by the *Access Enabler* across all content developers to represent the AuthN for that MVPD subscriber. In addition, the Adobe Access Server and Token Verifier can be operated by the CDN or a service provider on behalf of the content provider.