Widevine Java APIs for Android OS DRM Framework

Widevine Technologies
901 Fifth Ave, Suite 3400
Seattle, WA 98164 USA
www.widevine.com

206.254.3000 voice
206.254.3001 fax
sales@widevine.com

WIDEVINE®

1 Revision History

11/15/2010	1.0	Initial version	Edwin Wong
11/18/2010	1.1	Updated to changes from https://review.source.android.com/#change,18951; added acquireRights and getMetadata, acquireDrmInfo is no longer an asynchronous function and updated processDrmInfo sections	Edwin Wong
11/19/2010	1.2	Added introductory section, reformatted tables and reordered sections. Removed some unused methods.	Jeff Tinker
11/23/2010	1.3	Updated getConstraints	Edwin Wong
11/29/2010	1.4	Updated the following sections: processDrmInfo: rights information is stored in secure store, not returned by the API acquireRights: does not return rights information checkRightsStatus: only supports DEFAULT and PLAY actions	Edwin Wong
12/06/2010	1.5	Clarifies return values for getConstraints Updated ErrorEvent return values for onAcquireRights, onAcquireDrmInfo and onProcessDrmInfo	Edwin Wong
12/9/2010	1.6	Clarified that acquireDrmInfo must be called prior to using several methods.	Jeff Tinker
12/11/2010	1.7	Corrected online/offline description in processDrmInfo.	Jeff Tinker

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

Table of Contents

© 2010 Widevine Technologies, Inc. All Rights Reserved. Widevine, Widevine Cypher, Cypher Virtual SmartCard, Cypher VOD, Cypher Broadcast, Cypher for the PC, Widevine Digital Copy Protection, Widevine ZapTrack, Widevine TurboZap, Widevine Mensor and Widevine MediaProtect are either registered trademarks or trademarks of Widevine Technologies, Inc. and its subsidiaries in the United States and/or other countries. All other trademarks and trade names are the property of their respective owners. No express or implied warranties are provided for herein. All specifications are subject to change and any expected future products, features or functionality will be provided on an if and when available basis. Widevine reserves the right to substitute hardware component vendors and quantities in order to meet the customer specific environment and based on component availability. Note that the descriptions of Widevine Technologies' patents and other intellectual property herein are intended to provide illustrative, non-exhaustive examples of some of the areas to which the patents and applications are currently believed to pertain, and is not intended for use in a legal proceeding to interpret or limit the scope or meaning of the patents or their claims, or indicate that a Widevine patent claim(s) is materially required to perform or implement any of the preceding listed items.

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

2 INTRODUCTION

This document defines the Java APIs for Widevine DRM Plugin on Google Android devices. The Widevine DRM Plugin is part of the Android open DRM framework. Function parameters, error codes, return types, event handlers and error handlers for the APIs are described in this document.

The DRM APIs allow a client application to acquire a license for DRM protected content and manage the lifecycle of the license.

Figure 1 - Initial License Acquisition shows the process for acquiring a DRM license.

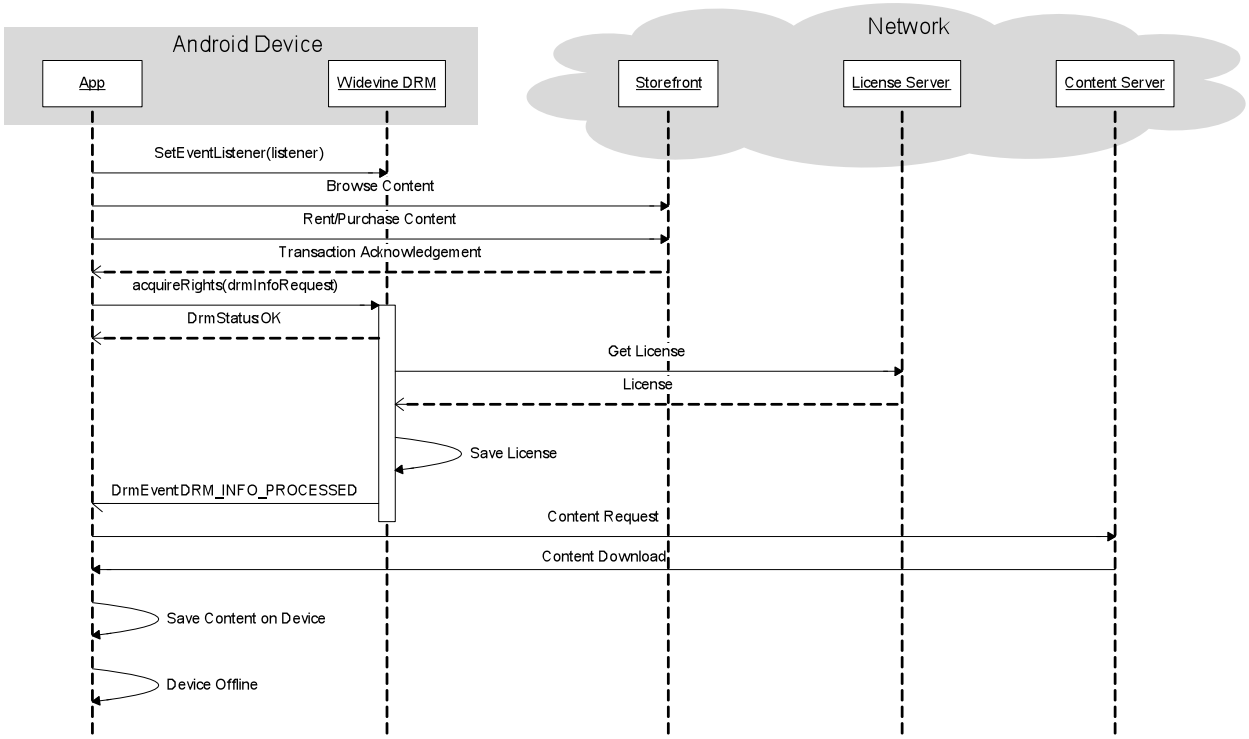

FIGURE 1 - INITIAL LICENSE ACQUISITION

The application should first register its event, error and info listener objects using the DrmManagerClient setOnListener methods. Several of the DRM APIs are asynchronous and will send events, info and errors to these listeners.

When a user browses content and performs a rental or purchase transaction, the application should construct a DrmInfoRequest object representing the license request

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

and providing essential parameters, and then invoke the acquireRights method. The license will be requested from the License Server and will be stored locally on the device after it is received. A DrmEvent of type DRM_INFO_PROCESSED will be generated to inform the application that the license is available. Once the license has been received, the content can be played.

To synchronize the rights status after a device has been offline, the application should call checkRightsStatus when the device is reconnected and becomes online. This scenario is shown in Figure 2 - License Refresh.

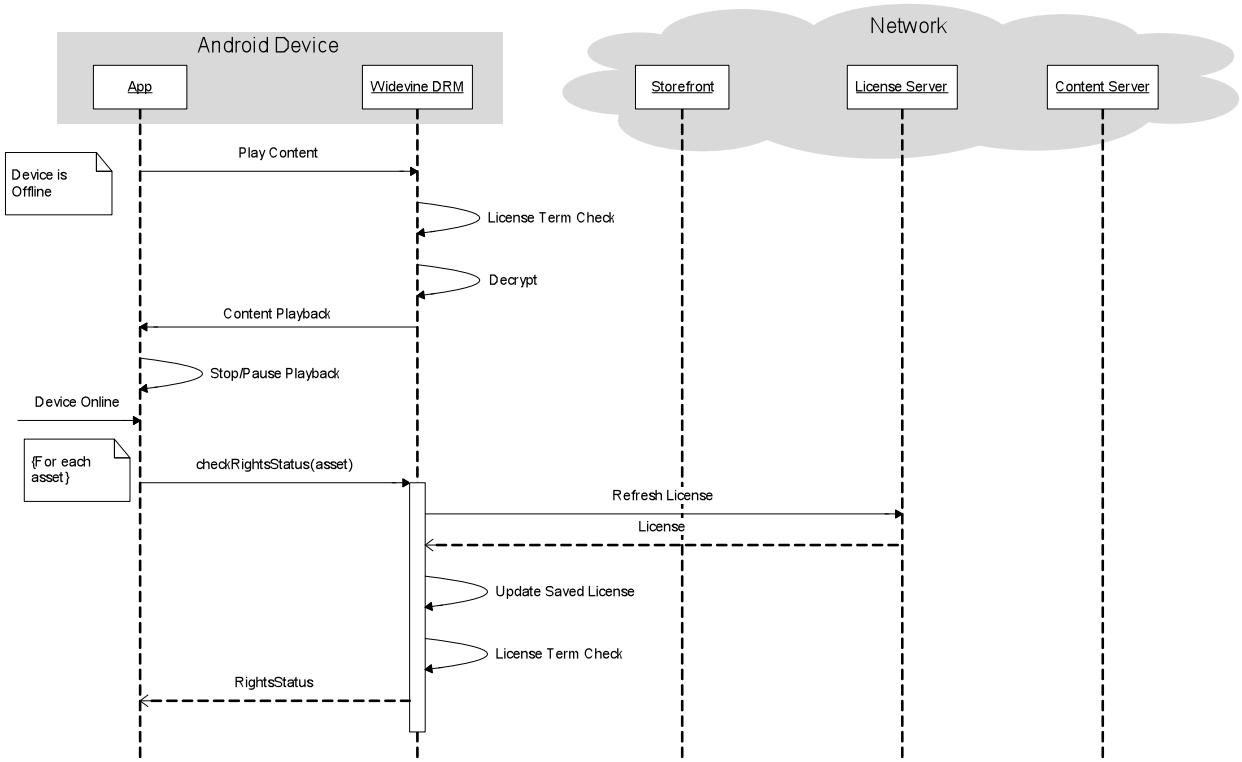

FIGURE 2 - LICENSE REFRESH

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

3 DrmManagerClient.acquireRights

Performs license acquisition and retrieves rights for the protected content from DRM server. This API essentially calls acquireDrmInfo and then calls processDrmInfo.

This is an asynchronous API. Result of the transaction will be retrieved from either the onEventListener or the onErrorListener event handler.

Parameters:

[in] drmInfoRequest - DrmInfoRequest class[1]

Returns:

ERROR_NONE for success, ERROR_UNKNOWN for failure;
Widevine error status is returned via onErrorListener event.

Events:

DrmEvent.TYPE_DRM_INFO_PROCESSED is the event generated upon completion of the API. The application calls DrmManagerClient.setOnEventListener() to set up an event handler.

Rights information is stored in a secure store and is not returned by this API.

Error Events:

The application calls mDrmManager.setOnErrorListener() to set up an error event handler.

DrmErrorEvent.TYPE_PROCESS_DRM_INFO_FAILED is returned.

[1] See next section for a description of DrmInfoRequest class.

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

3.1 About DrmInfoRequest Class

This entity class specifies parameters which defines the kind of information to retrieve from DRM server.

Parameters:

[in] infoType - specifies what kind of DRM information to retrieve

Supported type is: DrmInfoRequest.TYPE_RIGHTS_ACQUISITION_INFO

The TYPE_RIGHTS_ACQUISITION_INFO request defines parameters used during entitlement request as well as local playback. Use the put() method to initialize the HashMap<Key String, Object> pairs, which are described in the table below.

Key	*Object*	*Description*
WVDRMServerKey	String	URL for the CA server CGI.
WVAssetURIKey	String	File path or URI to the asset. Metadata from the content indicated by the AssetURI will be read to obtain {AssetId, SystemId and KeyId} if these values are not specified. The AssetURI is also used as a key to index the local license store.
WVAssetDBPathKey	String	Optional. File system path of the directory where the asset database should be stored. If specified, it overrides the default.
WVPortalKey	String	Mandatory. Contains portal value to be passed along to the CA system.
WVCAUserDataKey	String	Optional. Contains “userdata” to be passed along to CA system.
WVDeviceIDKey	String	Optional. Contains the device ID to be passed along to the CA system. This uniquely identifies an Android device.
WVStreamIDKey	String	Optional. Contains the stream ID to be passed along to the CA system. This identifies a playback attempt.
WVAssetIDKey	Integer	Optional. AssetI ID assigned to this content during the encryption process. If not specified, the Asset ID will be obtained by accessing the content metadata.
WVSystemIDKey	Integer	Optional. System ID assigned during the encryption process. If not specified, the System ID will be obtained by accessing the content metadata.
WVKeyIDKey	Integer	Optional. Key ID assigned during the encryption process. If not specified, the Key ID will be obtained by accessing the content metadata.

[in] mimeType - MIME type

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

Widevine DRM plugin currently supports “video/wvm” as the mimeType.

Example:

```
private static class Settings {
        public static String WIDEVINE_MIME_TYPE = "video/wvm";
        public static String DRM_SERVER_URI =
"http://wstfcps005.shibboleth.tv/widevine/cypherpc/cgi-bin/GetEMMs.cgi";
        public static String DEVICE_ID = "DEV12345";         // sample, do use a unique ID
        public static String PORTAL_NAME = "YouTube";        // identifies the portal
        public static String USER_DATA = "";                 // available for application use
};

DrmInfo rightsAcquisitionInfo = new DrmInfo(DrmInfoRequest.TYPE_RIGHTS_ACQUISITION_INFO,
        Settings.WIDEVINE_MIME_TYPE);

rightsAcquisitionInfo.put("WVAssetURIKey", assetUri.toString());
rightsAcquisitionInfo.put("WVDRMServerKey",  Settings.DRM_SERVER_URI);
rightsAcquisitionInfo.put("WVDeviceIDKey",  Settings.DEVICE_ID)
rightsAcquisitionInfo.put("WVPortalKey", Settings.PORTAL_NAME);

mDrmManager.acquireRights(rightsAcquisitionInfo);
```


This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

4 DrmManagerClient.acquireDrmInfo

Returns an instance of DrmInfo; initializes parameters used during entitlement request as well as local playback. The acquireDrmInfo/processDrmInfo pair of methods may be used in place of acquireRights if more control over the license acquisition is desired.

public DrmInfo acquireDrmInfo(DrmInfoRequest drmInfoRequest)

Parameters:

[in] drmInfoRequest - DrmInfoRequest class

Returns:

DrmInfo instance if succeeded, NULL if failed

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

5 DrmManagerClient.processDrmInfo

Performs license acquisition and retrieves rights for the protected content from DRM server.

For online playback, it is not necessary to invoke processDrmInfo, it is sufficient to only call acquireDrmInfo. For offline playback of locally stored content however, processDrmInfo or acquireRights must be called to acquire and store the DRM rights on the device prior to taking the device offline.

This is an asynchronous API. Result of the transaction will be retrieved from either the onEventListener or the onErrorListener event handler.

public int processDrmInfo(DrmInfo drmInfo)

Parameters:

[in] drmInfo - DrmInfo class[2]

The DrmManagerClient.acquireDrmInfo API can be used to create a DrmInfo instance, which is then passed as the argument to the processDrmInfo API.

Returns:

ERROR_NONE for success, ERROR_UNKNOWN for failure;
Widevine error status is returned via onErrorListener event.

Events:

DrmEvent.TYPE_DRM_INFO_PROCESSED is the event generated upon completion of the API. The application calls DrmManagerClient.setOnEventListener() to set up an event handler.

Rights information is stored in a secure store and is not returned by this API.

Error Events:

The application calls mDrmManager.setOnErrorListener() to set up an error event handler.

DrmErrorEvent.TYPE_PROCESS_DRM_INFO_FAILED is returned.

[2] See next section for a description of DrmInfo class.

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

5.1 About DrmInfo Class

The DrmInfo class constructor takes three parameters, i.e. DrmInfo(int infoType, const DrmBuffer& drmBuffer, const String8& mimeType).

Parameters:

[in] infoType - specifies what kind of DRM action to perform

Supported type is: DrmInfoRequest.TYPE_RIGHTS_ACQUISITION_INFO

[in] drmBuffer - data in drmBuffer is defined by infoType

- TYPE_RIGHTS_ACQUISITION_INFO

  drmBuffer contains URL for the CA server CGI program, e.g. http://wstfcps005.shibboleth.tv/widevine/cypherpc/cgi-bin/GetEMMs.cgi. This parameter is optional; it overrides the URL coded into the assets if specified.

The following parameters are passed in as HashMap<Key String, Object> attributes:

Key	*Object*	*Description*
WVAssetURIKey	String	Mandatory. File path or URI to the asset. Metadata from the content indicated by the AssetURI will be read to obtain {AssetId, SystemId and KeyId} if these values are not specified. The AssetURI is also used as a key to index the local license store.
WVAssetDBPathKey	String	Optional. File system path of the directory where the asset database should be stored. If specified, it overrides the default location.
WVPortalKey	String	Mandatory. Contains portal value to be passed along to the CA system.
WVCAUserDataKey	String	Optional. Contains “userdata” to be passed along to CA system.
WVDeviceIDKey	String	Optional. Contains the device ID to be passed along to the CA system. This uniquely identifies an Android device.
WVStreamIDKey	String	Optional. Contains the stream ID to be passed along to the CA system. This identifies a playback attempt.

[in] mimeType - MIME type

Widevine DRM plugin currently supports “video/wvm” as the mimeType.

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

6 DrmManagerClient.canHandle

Checks whether the given content URI, or content path, locates content that contains supported mimeType.

```
public boolean canHandle(String path, String mimeType)

public boolean canHandle(Uri uri, String mimeType)
```

Parameters:

[in] path/uri - content path or content URI

[in] mimeType - MIME type

Widevine DRM plugin currently supports “video/wvm” as the mimeType.

Returns:

True if content URI or content path locates content that uses “video/wvm” MIME type; otherwise, this function returns false.

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

7 DrmManagerClient.getOriginalMimeType

Retrieves the embedded MIME type from the original content.

```
public String getOriginalMimeType(String path)

public String getOriginalMimeType(Uri uri)
```

Parameters:

[in] path/uri - content path or content URI

Returns:

A “video/wm” string as the supported MIME type.

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

8 DrmManagerClient.checkRightsStatus

Checks whether the content has valid rights for a given action. Causes the license to be refreshed from the server. If the action parameter is omitted, DrmStore.Action.DEFAULT is assumed.

You must call DrmManagerClient.acquireDrmInfo (or DrmManagerClient.acquireRights) for the asset indicated by *path* prior to using this method.

```
public int checkRightsStatus(String path)

public int checkRightsStatus(String path, int action)

public int checkRightsStatus(Uri uri)

public int checkRightsStatus(Uri uri, int action)
```

Parameters:

[in] path/uri - content path or content URI

[in] action - specifies the action that can be performed on protected content

Supported actions are:

DrmStore.Action.DEFAULT
DrmStore.Action.PLAY

Returns:

An integer value indicating whether the content has valid rights.

The integer value is defined in DrmStore.RightsStatus as:

DrmStore.RightsStatus.RIGHTS_VALID
DrmStore.RightsStatus.RIGHTS_INVALID
DrmStore.RightsStatus.RIGHTS_EXPIRED
DrmStore.RightsStatus.RIGHTS_NOT_ACQUIRED

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

9 DrmManagerClient.removeRights

Removes the rights associated with the given protected content.

You must call DrmManagerClient.acquireDrmInfo (or DrmManagerClient.acquireRights) for the asset indicated by *path* prior to using this method.

```
public int removeRights(String path)

public int removeRights(Uri uri)
```

Parameters:

[in] path/uri - content path or content URI

Returns:

ERROR_NONE for success, ERROR_UNKNOWN for failure

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

10 DrmManagerClient.removeAllRights()

Removes all the rights information saved on the device. This API will be used in master reset.

This is an asynchronous API. Result of the transaction will be retrieved from either the onEventListener or the onErrorListener event handler.

You must call DrmManagerClient.acquireDrmInfo (or DrmManagerClient.acquireRights) for at least one asset prior to using this method, to set the portal value.

```
public int removeAllRights()
```

Returns:

ERROR_NONE for success, ERROR_UNKNOWN for failure

Event:

DrmEvent.TYPE_ALL_RIGHTS_REMOVED is returned upon completion of the API. The application calls mDrmManager.setOnEventListener() to set up an event handler.

Error Event:

DrmErrorEvent. TYPE_REMOVE_ALL_RIGHTS _FAILED. The API is returned if the API failed to remove all rights saved on the device. The application calls mDrmManager.setOnErrorListener() to set up an error event handler.

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

11 DrmManagerClient.getConstraints

Gets constraints information evaluated from DRM content.

You must call DrmManagerClient.acquireDrmInfo (or DrmManagerClient.acquireRights) for the asset indicated by *path* prior to using this method.

```
public ContentValues getConstraints(String path, int action)

public ContentValues getConstraints(Uri uri, int action)
```

Parameters:

[in] path/uri - content path or content URI
[in] action - specifies the action that can be performed on protected content

Supported actions are:

DrmStore.Action.DEFAULT
DrmStore.Action.PLAY

Returns:

ContentValues instance in which constraints key-value pairs are embedded, NULL in case of failure

Valid keys returned by this API are:

DrmStore.ConstraintsColumns.LICENSE_START_TIME
DrmStore.ConstraintsColumns.LICENSE_EXPIRY_TIME
DrmStore.ConstraintsColumns.LICENSE_AVAILABLE_TIME

The following parameters are returned in HashMap<Key String, Object> attributes:

Key	*Object*	*Description*
LICENSE_START_TIME	String	Seconds elapsed since the start of playback
LICENSE_EXPIRY_TIME	String	Seconds left till license expiration once playback has started
LICENSE_AVAILABLE_TIME	String	Time remaining in seconds for user to complete playback (e.g. a 24 hour window)

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.

12 DrmManagerClient.getDrmObjectType

Retrieves the type of the protected object (content, rights, etc…) using specified uri or mimeType. At least one parameter should be non-null to retrieve DRM object type.

```
public int getDrmObjectType(String path, String mimeType)

public int getDrmObjectType(Uri uri, String mimeType)
```

Parameters:

[in] path/uri - object path or object URI

[in] mimeType - MIME type

Widevine DRM plugin currently supports “video/wvm” as the mimeType.

Returns:

This API is not supported by Widevine plugin. The function always returns DrmStore.DrmObjectType.UNKNOWN.

This document contains confidential information and is proprietary to Widevine Technologies Inc., and may not be reproduced or redistributed without the express written permission of Widevine Technologies.