PROD-5130 Verimatrix VCAS:
Common Encoder Interface API Reference

The information contained in this document is provided AS-IS without warranty of any kind. Verimatrix reserves the right to make changes to this document and its implementation without any obligation to notify the recipient of this document. Nothing in this document shall create a warranty, either express or implied, and nothing herein shall alter the terms and conditions set forth in the applicable confidentiality, license, and/or service agreement(s) for use of the Verimatrix confidential information, products, or services.

All of the features described in this document may not be currently available. Refer to the most recent product announcement or contact Verimatrix, Inc. for information on feature and product availability.

Verimatrix, Inc.
6059 Cornerstone Court West
San Diego, CA 92121-3713
Telephone: +1.858.677.7800
Fax: +1.858.677.7804

Table of Contents

Preface

About This Document

This document provides a description of the application program interface (API) between encoders/scramblers and the Verimatrix VCAS system.

The Verimatrix VCAS Common Encoder Interface (CEI) has two encoder interfaces:

- Native
- DASH-IF Content Protection Information Exchange (CPIX)

This document describes the part of the CEI API that defines the key exchange with encoders using the native specification. This document describes both CEI v2.0 and CEI v1.1.

For the CEI CPIX API, refer to *PROD-5132 Verimatrix VCAS: Common Encoder Interface CPIX API Reference*.

Who Should Use This Document

Users of this document are those trying to integrate their encoders/scramblers to the Verimatrix VCAS system.

Readers should have:

- Working knowledge of encryption and key concepts
- Understanding of encoder/scrambler functionality
- Understanding of video formats and processing
- General knowledge of the Verimatrix VCAS system
- Have attended Verimatrix University

What Changed in This Document

The following information has been added or revised for this version of the document.

SECTION	SOFTWARE VERSION	DESCRIPTION OF CHANGE
ALL	4.3.0.0	Footers updated to 4.3
Create/Retrieve Key ContentInfo	4.3.3.0	Clarified that CEI supports only a single key request/retrieve
Available Methods v2.0	4.3.3.0	Added section
Error Codes	4.3.3.0	Added section
AssetKey	4.3.3.0	Added section
CreateAssetResponse	4.3.3.0	Added section
ContentKey	4.3.3.0	Added information about CEI version
Overview	4.3.3.0	Added communication information for v2.0
Create Asset	4.3.4.1	Updated for new `position` parameter
All	4.3.5.0	New Verimatrix product names (see below)

New Verimatrix Product Names

Since February 2020, several Verimatrix products names have changed. The following table shows the changes.

OLD VERIMATRIX PRODUCT NAME	NEW VERIMATRIX PRODUCT NAME
VCAS™ and VCAS Ultra™	Verimatrix VCAS
MultiRights™	Multi-DRM
VideoMark™ and StreamMark™	Verimatrix Watermarking
RightsConnex™	Content Distribution
nTitleMe™	TV Everywhere Authentication
Verspective™	Verimatrix Analytics

Related Documents

In addition to this document, the following documents may also be useful when using Verimatrix VCAS:

- *VCAS Multi-network Revenue Security – Overview.* Provides an overview of Verimatrix VCAS Platform and the available Verimatrix VCAS solutions. This document can be found on the Verimatrix Partner Portal.

- *PROD-5120 Verimatrix VCAS for Internet TV: Integration Guide*. Provides configuration information for the Verimatrix VCAS for Internet TV solution.

- *PROD-5132 Verimatrix VCAS: Common Encoder Interface CPIX API Reference*. Defines the available methods for the CEI CPIX API.

- *PROD-5230 Verimatrix VCAS for Multi-DRM: Configuration Guide*. Provides configuration information for the Verimatrix VCAS for Multi-DRM solution.

- *PROD-5231 Verimatrix VCAS for Multi-DRM PlayReady: Configuration Guide*. Provides configuration information for the Verimatrix VCAS for Multi-DRM PlayReady solution.

Chapter 1

Overview

The Common Encoder Interface (CEI) is a RESTful interface using JavaScript Object Notation (JSON) format for the request body.

The primary functionality of the CEI is:

- Protection of content with multiple digital rights management (DRM) schemes (simultaneously)
- Support multiple keys per content item
- Support for content policies (copy controls, offline playback, and others)

Communication between an encoder and the CEI v2.0 occurs as follows:

1. Encoder requests the creation of a new content item using the Create Asset (on page 3) request.
2. Encoder requests creation of a new key using the Create Key (on page 6) request.
3. Encoder requests a content key using one of the following methods:
   - Get Key by assetId and position (on page 12)
   - Get Key by assetId and keyId (on page 17)
   - Get Key by keyId (on page 22)
4. Optional. Encoder sends a periodic Heartbeat (on page 26) request to ensure that the connection between the encoder and the CEI is still alive.

Communication between an encoder and the CEI v1.1 occurs as follows:

5. Encoder requests the creation of a new content item using the Create Content (on page 28) request. In the scope of the Verimatrix VCAS deployment, the content is uniquely identified by the `contentId`. In the case of multiple Verimatrix VCAS deployments, an optional deployment specific `siteId` can be prepended to the `contentId`. The pair, `siteId:contentId`, constitutes a unique content identifier across all the Verimatrix VCAS deployments.

6. Encoder requests a content key using the Create/Retrieve Key (on page 30) request. The encoder must include the `contentId` and the `timestamp` in the request. The response to the Create/Retrieve key request includes the `key`, `key ID`, and the metadata specific to each DRM enabled on the Verimatrix VCAS system. The encoder can issue multiple key requests per content item; each with a different `timestamp.`

7. Optional. Encoder sends a periodic Heartbeat (on page 37) request to ensure that the connection between the encoder and the CEI is still alive.

Chapter 2

Available Methods: CEI v2.0

This section describes the methods available in the CEI v2.0 API.

URI Scheme

BasePath: /cei/v2.0

Create Asset

Creates a content item based on the passed-in parameters. In the scope of the Verimatrix VCAS deployment, the `assetId` uniquely defines the content. In the case of multiple Verimatrix VCAS deployments, the encoder can optionally prepend a deployment specific `siteId` to the `assetId`. The pair, `siteId:assetId`, constitutes a unique content identifier across all the Verimatrix VCAS deployments.

An encoder must send this request before sending any `Create Key` request for a content item.

Request Path

```
POST /cei/v2.0/assets?siteId={siteId}&position={position}
```

Request Body

```
{
    "createAssetRequest" : {
        "assetId": string,
        "assetType": string
    }
}
```

Request Example

```
curl -v -X POST 'http://localhost:8058/cei/v2.0/assets?siteId=1&position=1' -
H 'Content-Type: application/json'
-d '{
  "createAssetRequest" : {
     "assetId": "test",
    "assetType": "DTV"
  }
}'
```

Request Parameters

NAME	TYPE	OCCURS	DESCRIPTION
`siteId`	Integer	0..1	**Optional**. Unique identifier for the site. Use a site identifier to differentiate between operators. If you do not provide a `siteId`, the method uses the local `siteId` from the license. Example: 121
`position`	Long	0..1	**Optional.** Epoch time associated with the start time of the asset. `position` applies only to DTV assets. If added to a VOD asset, it is ignored. Default: Current epoch time Valid range: 0 – 2147483647 (00:00:00 1-Jan-1970 - 03:14:07 19-Jan-2038 UTC)
`assetId`	String	1..1	Unique identifier for the content. Example: test Default: none
`assetType`	String	1..1	Content delivery type. Valid values: • VOD • DTV

Response Example

```
HTTP/1.1 200 OK
{
    "createAssetResponse": {
        "status":"OK",
        "statusMessage":"Successfully created asset with assetId=test and
siteId=1"
    }
}
```

Response Parameters

HTTP CODE	TYPE	OCCURS	DESCRIPTION
200	createAssetResponse (on page 39)	1..1	Response object containing status information

For information on other possible HTTP responses, refer to Error Codes (on page 47).

Create Key

Creates a key in a specified position for an existing content item.

Request Path

```
POST /cei/v2.0/assets/{assetId}/keys?siteId={siteId}
```

Request Body

```
{
    "createKeyRequest" : {
        "position": long
    }
}
```

Request Parameters

NAME	TYPE	OCCURS	DESCRIPTION
`assetId`	String	1..1	Unique identifier for the content item. `assetId` is the same as the `assetId` that you defined during content item creation. For more information, refer to Create Asset (on page 3). Example: test
`position`	Long	1..1	Epoch time or index of the position. `position` must not be a negative number. If the key is for Internet TV DTV content then `position` must be epoch time. Epoch time must be in the future. Examples: • 1458172978 • 1
`siteId`	Integer	0..1	**Optional**. Unique identifier for the site. Use a site identifier to differentiate between operators. If you do not provide a `siteId`, the method uses the local `siteId` from the license. Example: 1

Request Example

```
curl -v -X POST 'http://localhost:8058/cei/v2.0/assets/test/keys?siteId=1' -H
'Content-Type: application/json'
 -d '{"createKeyRequest": {"position": 1 }}'
```

Response Example

```
HTTP/1.1 200 OK
{
  "keyResponse": {
    "status": "OK",
    "statusMessage": "Successfully created key for assetId=test, position=1,
siteId=1",
    "drmList": [
      {
        "drmName": "PLAYREADY",
        "streamingProtocol": "SS",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "position": 1
            },
            "drmData": {
              "manifestHeader":
"AgMAAAEAAQD4AjwAVwBSAE0ASABFAEEARABFAFIAIAB4AG0AbABuAHMAPQAiAGgAdAB0AHAAOgAv
AC8AcwBjAGgAZQBtAGEAcwAuAG0AaQBjAHIAbwBzAG8AZgB0AC4AYwBvAG0ALwBEAFIATQAvADIAM
AAwADcALwAwADMALwBQAGwAYQB5AFIAZQBhAGQAeQBIAGUAYQBkAGUAcgAiACAAdgBlAHIAcwBpAG
8AbgA9ACIANAAuADAALgAwAC4AMAAiAD4APABEAEEAVABBAD4APABQAFIATwBUAEUAQwBUAEkATgB
GAE8APgA8AEsARQBZAEwARQBOAD4AMQA2ADwALwBLAEUAWQBMAEUATgA+ADwAQQBMAEcASQBEAD4A
QQBFAFMAQwBUAFIAPAAvAEEATABHAEkARAA+ADwALwBQAFIATwBUAEUAQwBUAEkATgBGAE8APgA8A
EsASQBEAD4ASQBCAEoAdAAvADkAZABsAE8AWQAyAEEAVABOADYAMAAzAEsATwBOAGQAZwA9AD0APA
AvAEsASQBEAD4APABDAEgARQBDAEsAUwBVAE0APgA0AFcAbQA4AHgAVwA4AHEARQBrAEEAPQA8AC8
AQwBIAEUAQwBLAFMAVQBNAD4APABMAEEAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC8AMQAyADcALgAw
AC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQBuAGEAZwBlAHIAL
gBhAHMAbQB4ADwALwBMAEEAXwBVAFIATAA+ADwATABVAEkAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC
8AMQAyADcALgAwAC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQB
uAGEAZwBlAHIALgBhAHMAbQB4ADwALwBMAFUASQBfAFUAUgBMAD4APAAvAEQAQQBUAEEAPgA8AC8A
VwBSAE0ASABFAEEARABFAFIAPgA="
            }
          }
        ]
      },
      {
        "drmName": "PLAYREADY",
        "streamingProtocol": "DASH",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "position": 1
            },
            "drmData": {
              "drmSystemId": "9a04f079-9840-4286-ab92-e65be0885f95",
              "pssh": {
                "data":
"AgMAAAEAAQD4AjwAVwBSAE0ASABFAEEARABFAFIAIAB4AG0AbABuAHMAPQAiAGgAdAB0AHAAOgAv
AC8AcwBjAGgAZQBtAGEAcwAuAG0AaQBjAHIAbwBzAG8AZgB0AC4AYwBvAG0ALwBEAFIATQAvADIAM
AAwADcALwAwADMALwBQAGwAYQB5AFIAZQBhAGQAeQBIAGUAYQBkAGUAcgAiACAAdgBlAHIAcwBpAG
8AbgA9ACIANAAuADAALgAwAC4AMAAiAD4APABEAEEAVABBAD4APABQAFIATwBUAEUAQwBUAEkATgB
GAE8APgA8AEsARQBZAEwARQBOAD4AMQA2ADwALwBLAEUAWQBMAEUATgA+ADwAQQBMAEcASQBEAD4A
QQBFAFMAQwBUAFIAPAAvAEEATABHAEkARAA+ADwALwBQAFIATwBUAEUAQwBUAEkATgBGAE8APgA8A
EsASQBEAD4ASQBCAEoAdAAvADkAZABsAE8AWQAyAEEAVABOADYAMAAzAEsATwBOAGQAZwA9AD0APA
AvAEsASQBEAD4APABDAEgARQBDAEsAUwBVAE0APgA0AFcAbQA4AHgAVwA4AHEARQBrAEEAPQA8AC8
AQwBIAEUAQwBLAFMAVQBNAD4APABMAEEAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC8AMQAyADcALgAw
AC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQBuAGEAZwBlAHIAL
gBhAHMAbQB4ADwALwBMAEEAXwBVAFIATAA+ADwATABVAEkAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC
8AMQAyADcALgAwAC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQB
uAGEAZwBlAHIALgBhAHMAbQB4ADwALwBMAFUASQBfAFUAUgBMAD4APAAvAEQAQQBUAEEAPgA8AC8A
VwBSAE0ASABFAEEARABFAFIAPgA="
              }
            }
          }
        ]
      },
      {
        "drmName": "WIDEVINE",
        "streamingProtocol": "DASH",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "position": 1
            },
            "drmData": {
              "drmSystemId": "edef8ba9-79d6-4ace-a3c8-27dcd51d21ed",
              "pssh": {
                "data":
"CAESEP9tEiBl1405gEzetNyjjXYaCnZlcmltYXRyaXgiCnI9dGVzdCZzPTEqBVNEX0hE"
              }
            }
          }
        ]
      },
      {
        "drmName": "FAIRPLAY_STREAMING",
        "streamingProtocol": "HLS",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "iv": "ee7166d46ddda0af473a4a1ad6e4463c",
              "position": 1
            },
            "drmData": {
              "licenseServerUrl": "http://sp-
mr433.dev.ussd.verimatrix.com:8064/fpsa/v1.0",
              "keyUri": "r=test&s=1"
            }
          }
        ]
      }
    ]
  }
}
```

Response Parameters

HTTP CODE	TYPE	OCCURS	DESCRIPTION
200	KeyResponse (on page 42)	1..1	Key response object containing information about the content keys for different DRM types.

For information on other possible HTTP responses, refer to Error Codes (on page 47).

Get Key by assetId and Position

Retrieves a key for a content item with a given `assetId` and key `position`.

Request Path

```
GET /cei/v2.0/assets/{assetId}/keys?position=1&siteId=1
```

Request Parameters

NAME	TYPE	OCCURS	DESCRIPTION
`assetId`	String	1..1	Unique identifier for the content item. `assetId` is the same as the `assetId` that was defined during creation of the content item. For more information, refer to Create Asset (on page 3). Example: test
`position`	Long	1..1	Epoch time or index for which the encoder is requesting the key. `position` is the same as the `position` that was defined during creation of the key. For more information, refer to Create Key (on page 6). Example: 1458172978
`siteId`	Integer	0..1	**Optional**. Unique identifier for the site. Use a site identifier to differentiate between operators. If you do not provide a `siteId`, the method uses the local `siteId` from the license. Example: 121

Request Example

```
curl -v -X GET
'http://localhost:8058/cei/v2.0/assets/test/keys?siteId=1&position=1'
```

Response Example

```
HTTP/1.1 200 OK
{
  "keyResponse": {
    "status": "OK",
    "statusMessage": "Successfully retrieved key for assetId=test,
position=1, siteId=1",
    "drmList": [
      {
        "drmName": "PLAYREADY",
        "streamingProtocol": "SS",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "position": 1
            },
            "drmData": {
              "manifestHeader":
"AgMAAAEAAQD4AjwAVwBSAE0ASABFAEEARABFAFIAIAB4AG0AbABuAHMAPQAiAGgAdAB0AHAAOgAv
AC8AcwBjAGgAZQBtAGEAcwAuAG0AaQBjAHIAbwBzAG8AZgB0AC4AYwBvAG0ALwBEAFIATQAvADIAM
AAwADcALwAwADMALwBQAGwAYQB5AFIAZQBhAGQAeQBIAGUAYQBkAGUAcgAiACAAdgBlAHIAcwBpAG
8AbgA9ACIANAAuADAALgAwAC4AMAAiAD4APABEAEEAVABBAD4APABQAFIATwBUAEUAQwBUAEkATgB
GAE8APgA8AEsARQBZAEwARQBOAD4AMQA2ADwALwBLAEUAWQBMAEUATgA+ADwAQQBMAEcASQBEAD4A
QQBFAFMAQwBUAFIAPAAvAEEATABHAEkARAA+ADwALwBQAFIATwBUAEUAQwBUAEkATgBGAE8APgA8A
EsASQBEAD4ASQBCAEoAdAAvADkAZABsAE8AWQAyAEEAVABOADYAMAAzAEsATwBOAGQAZwA9AD0APA
AvAEsASQBEAD4APABDAEgARQBDAEsAUwBVAE0APgA0AFcAbQA4AHgAVwA4AHEARQBrAEEAPQA8AC8
AQwBIAEUAQwBLAFMAVQBNAD4APABMAEEAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC8AMQAyADcALgAw
AC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQBuAGEAZwBlAHIAL
gBhAHMAbQB4ADwALwBMAEEAXwBVAFIATAA+ADwATABVAEkAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC
8AMQAyADcALgAwAC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQB
uAGEAZwBlAHIALgBhAHMAbQB4ADwALwBMAFUASQBfAFUAUgBMAD4APAAvAEQAQQBUAEEAPgA8AC8A
VwBSAE0ASABFAEEARABFAFIAPgA="
```

```
            }
          }
        ]
      },
      {
        "drmName": "PLAYREADY",
        "streamingProtocol": "DASH",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "position": 1
            },
            "drmData": {
              "drmSystemId": "9a04f079-9840-4286-ab92-e65be0885f95",
              "pssh": {
                "data":
"AgMAAAEAAQD4AjwAVwBSAE0ASABFAEEARABFAFIAIAB4AG0AbABuAHMAPQAiAGgAdAB0AHAAOgAv
AC8AcwBjAGgAZQBtAGEAcwAuAG0AaQBjAHIAbwBzAG8AZgB0AC4AYwBvAG0ALwBEAFIATQAvADIAM
AAwADcALwAwADMALwBQAGwAYQB5AFIAZQBhAGQAeQBIAGUAYQBkAGUAcgAiACAAdgBlAHIAcwBpAG
8AbgA9ACIANAAuADAALgAwAC4AMAAiAD4APABEAEEAVABBAD4APABQAFIATwBUAEUAQwBUAEkATgB
GAE8APgA8AEsARQBZAEwARQBOAD4AMQA2ADwALwBLAEUAWQBMAEUATgA+ADwAQQBMAEcASQBEAD4A
QQBFAFMAQwBUAFIAPAAvAEEATABHAEkARAA+ADwALwBQAFIATwBUAEUAQwBUAEkATgBGAE8APgA8A
EsASQBEAD4ASQBCAEoAdAAvADkAZABsAE8AWQAyAEEAVABOADYAMAAzAEsATwBOAGQAZwA9AD0APA
AvAEsASQBEAD4APABDAEgARQBDAEsAUwBVAE0APgA0AFcAbQA4AHgAVwA4AHEARQBrAEEAPQA8AC8
AQwBIAEUAQwBLAFMAVQBNAD4APABMAEEAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC8AMQAyADcALgAw
AC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQBuAGEAZwBlAHIAL
gBhAHMAbQB4ADwALwBMAEEAXwBVAFIATAA+ADwATABVAEkAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC
8AMQAyADcALgAwAC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQB
uAGEAZwBlAHIALgBhAHMAbQB4ADwALwBMAFUASQBfAFUAUgBMAD4APAAvAEQAQQBUAEEAPgA8AC8A
VwBSAE0ASABFAEEARABFAFIAPgA="
              }
            }
          }
        ]
      },
      {
        "drmName": "WIDEVINE",
        "streamingProtocol": "DASH",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "position": 1
            },
            "drmData": {
              "drmSystemId": "edef8ba9-79d6-4ace-a3c8-27dcd51d21ed",
              "pssh": {
                "data":
"CAESEP9tEiBl1405gEzetNyjjXYaCnZlcmltYXRyaXgiCnI9dGVzdCZzPTEqBVNEX0hE"
              }
            }
          }
        ]
      },
      {
        "drmName": "FAIRPLAY_STREAMING",
        "streamingProtocol": "HLS",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "iv": "ee7166d46ddda0af473a4a1ad6e4463c",
              "position": 1
            },
            "drmData": {
              "licenseServerUrl": "http://sp-
mr433.dev.ussd.verimatrix.com:8064/fpsa/v1.0",
              "keyUri": "r=test&s=1"
            }
          }
        ]
      }
    ]
  }
}
```

Response Parameters

HTTP Code	Type	Occurs	Description
200	KeyResponse (on page 42)	1..1	Key response object containing information about the content keys for different DRM types.

For information on other possible HTTP responses, refer to Error Codes (on page 47).

Get Key by assetId and keyId

Retrieves a key for a content item with a given `assetId` and `keyId`.

Request Path

```
GET /cei/v2.0/assets/{assetId}/keys/{keyId}?siteId=1
```

Request Parameters

NAME	TYPE	OCCURS	DESCRIPTION
`assetId`	String	1..1	Unique identifier for the content item. `assetId` is the same as the `assetId` that you defined during content item creation. For more information, refer to Create Asset (on page 3). Example: test
`keyId`	Long	1..1	Unique key identifier in UUID format. `keyId` is the same as the `keyId` returned by the CEI during key creation. For more information, refer to Create Key (on page 6). Example: ff6d1220-65d7-8d39-804c-deb4dca38d76
`siteId`	Integer	0..1	**Optional**. Unique identifier for the site. Use a site identifier to differentiate between operators. If you do not provide a `siteId`, the method uses the local `siteId` from the license. Example: 121

Request Example

```
curl -v -X GET 'http://localhost:8058/cei/v2.0/assets/test/keys/ff6d1220-
65d7-8d39-804c-deb4dca38d76?siteId=1'
```

Response Example

```
HTTP/1.1 200 OK
{
  "keyResponse": {
    "status": "OK",
    "statusMessage": "Successfully retrieved key for assetId=test,
keyId=ff6d1220-65d7-8d39-804c-deb4dca38d76, siteId=1",
    "drmList": [
      {
        "drmName": "PLAYREADY",
        "streamingProtocol": "SS",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "position": 1
            },
            "drmData": {
              "manifestHeader":
"AgMAAAEAAQD4AjwAVwBSAE0ASABFAEEARABFAFIAIAB4AG0AbABuAHMAPQAiAGgAdAB0AHAAOgAv
AC8AcwBjAGgAZQBtAGEAcwAuAG0AaQBjAHIAbwBzAG8AZgB0AC4AYwBvAG0ALwBEAFIATQAvADIAM
AAwADcALwAwADMALwBQAGwAYQB5AFIAZQBhAGQAeQBIAGUAYQBkAGUAcgAiACAAdgBlAHIAcwBpAG
8AbgA9ACIANAAuADAALgAwAC4AMAAiAD4APABEAEEAVABBAD4APABQAFIATwBUAEUAQwBUAEkATgB
GAE8APgA8AEsARQBZAEwARQBOAD4AMQA2ADwALwBLAEUAWQBMAEUATgA+ADwAQQBMAEcASQBEAD4A
QQBFAFMAQwBUAFIAPAAvAEEATABHAEkARAA+ADwALwBQAFIATwBUAEUAQwBUAEkATgBGAE8APgA8A
EsASQBEAD4ASQBCAEoAdAAvADkAZABsAE8AWQAyAEEAVABOADYAMAAzAEsATwBOAGQAZwA9AD0APA
AvAEsASQBEAD4APABDAEgARQBDAEsAUwBVAE0APgA0AFcAbQA4AHgAVwA4AHEARQBrAEEAPQA8AC8
AQwBIAEUAQwBLAFMAVQBNAD4APABMAEEAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC8AMQAyADcALgAw
AC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQBuAGEAZwBlAHIAL
gBhAHMAbQB4ADwALwBMAEEAXwBVAFIATAA+ADwATABVAEkAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC
8AMQAyADcALgAwAC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQB
uAGEAZwBlAHIALgBhAHMAbQB4ADwALwBMAFUASQBfAFUAUgBMAD4APAAvAEQAQQBUAEEAPgA8AC8A
VwBSAE0ASABFAEEARABFAFIAPgA="
```

```
            }
          }
        ]
      },
      {
        "drmName": "PLAYREADY",
        "streamingProtocol": "DASH",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "position": 1
            },
            "drmData": {
              "drmSystemId": "9a04f079-9840-4286-ab92-e65be0885f95",
              "pssh": {
                "data":
"AgMAAAEAAQD4AjwAVwBSAE0ASABFAEEARABFAFIAIAB4AG0AbABuAHMAPQAiAGgAdAB0AHAAOgAv
AC8AcwBjAGgAZQBtAGEAcwAuAG0AaQBjAHIAbwBzAG8AZgB0AC4AYwBvAG0ALwBEAFIATQAvADIAM
AAwADcALwAwADMALwBQAGwAYQB5AFIAZQBhAGQAeQBIAGUAYQBkAGUAcgAiACAAdgBlAHIAcwBpAG
8AbgA9ACIANAAuADAALgAwAC4AMAAiAD4APABEAEEAVABBAD4APABQAFIATwBUAEUAQwBUAEkATgB
GAE8APgA8AEsARQBZAEwARQBOAD4AMQA2ADwALwBLAEUAWQBMAEUATgA+ADwAQQBMAEcASQBEAD4A
QQBFAFMAQwBUAFIAPAAvAEEATABHAEkARAA+ADwALwBQAFIATwBUAEUAQwBUAEkATgBGAE8APgA8A
EsASQBEAD4ASQBCAEoAdAAvADkAZABsAE8AWQAyAEEAVABOADYAMAAzAEsATwBOAGQAZwA9AD0APA
AvAEsASQBEAD4APABDAEgARQBDAEsAUwBVAE0APgA0AFcAbQA4AHgAVwA4AHEARQBrAEEAPQA8AC8
AQwBIAEUAQwBLAFMAVQBNAD4APABMAEEAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC8AMQAyADcALgAw
AC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQBuAGEAZwBlAHIAL
gBhAHMAbQB4ADwALwBMAEEAXwBVAFIATAA+ADwATABVAEkAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC
8AMQAyADcALgAwAC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQB
uAGEAZwBlAHIALgBhAHMAbQB4ADwALwBMAFUASQBfAFUAUgBMAD4APAAvAEQAQQBUAEEAPgA8AC8A
VwBSAE0ASABFAEEARABFAFIAPgA="
              }
            }
          }
        ]
      },
      {
        "drmName": "WIDEVINE",
        "streamingProtocol": "DASH",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "position": 1
            },
            "drmData": {
              "drmSystemId": "edef8ba9-79d6-4ace-a3c8-27dcd51d21ed",
              "pssh": {
                "data":
"CAESEP9tEiBl1405gEzetNyjjXYaCnZlcmltYXRyaXgiCnI9dGVzdCZzPTEqBVNEX0hE"
              }
            }
          }
        ]
      },
      {
        "drmName": "FAIRPLAY_STREAMING",
        "streamingProtocol": "HLS",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "iv": "ee7166d46ddda0af473a4a1ad6e4463c",
              "position": 1
            },
            "drmData": {
              "licenseServerUrl": "http://sp-
mr433.dev.ussd.verimatrix.com:8064/fpsa/v1.0",
              "keyUri": "r=test&s=1"
            }
          }
        ]
      }
    ]
  }
}
```

Response Parameters

HTTP CODE	TYPE	OCCURS	DESCRIPTION
200	KeyResponse (on page 42)	1..1	Key response object containing information about the content keys for different DRM types.

For information on other possible HTTP responses, refer to Error Codes (on page 47).

Get Key by keyId

Retrieves a key for a given `keyId`.

Request Path

```
GET /cei/v2.0/keys/{keyId}?siteId=1
```

Request Parameters

NAME	TYPE	OCCURS	DESCRIPTION
`keyId`	Long	1..1	Unique key identifier in UUID format. `keyId` is the same as the `keyId` returned by the CEI during key creation. For more information, refer to Create Key (on page 6). Example: ff6d1220-65d7-8d39-804c-deb4dca38d76
`siteId`	Integer	0..1	**Optional**. Unique identifier for the site. Use a site identifier to differentiate between operators. If you do not provide a `siteId`, the method uses the local `siteId` from the license. Example: 1

Request Example

```
curl -v -X GET 'http://localhost:8058/cei/v2.0/keys/ff6d1220-65d7-8d39-804c-
deb4dca38d76?siteId=1'
```

Response Example

```
HTTP/1.1 200 OK
{
  "keyResponse": {
    "status": "OK",
    "statusMessage": "Successfully retrieved key for keyId=ff6d1220-65d7-
8d39-804c-deb4dca38d76, siteId=1",
    "drmList": [
      {
        "drmName": "PLAYREADY",
        "streamingProtocol": "SS",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "position": 1
            },
            "drmData": {
              "manifestHeader":
"AgMAAAEAAQD4AjwAVwBSAE0ASABFAEEARABFAFIAIAB4AG0AbABuAHMAPQAiAGgAdAB0AHAAOgAv
AC8AcwBjAGgAZQBtAGEAcwAuAG0AaQBjAHIAbwBzAG8AZgB0AC4AYwBvAG0ALwBEAFIATQAvADIAM
AAwADcALwAwADMALwBQAGwAYQB5AFIAZQBhAGQAeQBIAGUAYQBkAGUAcgAiACAAdgBlAHIAcwBpAG
8AbgA9ACIANAAuADAALgAwAC4AMAAiAD4APABEAEEAVABBAD4APABQAFIATwBUAEUAQwBUAEkATgB
GAE8APgA8AEsARQBZAEwARQBOAD4AMQA2ADwALwBLAEUAWQBMAEUATgA+ADwAQQBMAEcASQBEAD4A
QQBFAFMAQwBUAFIAPAAvAEEATABHAEkARAA+ADwALwBQAFIATwBUAEUAQwBUAEkATgBGAE8APgA8A
EsASQBEAD4ASQBCAEoAdAAvADkAZABsAE8AWQAyAEEAVABOADYAMAAzAEsATwBOAGQAZwA9AD0APA
AvAEsASQBEAD4APABDAEgARQBDAEsAUwBVAE0APgA0AFcAbQA4AHgAVwA4AHEARQBrAEEAPQA8AC8
AQwBIAEUAQwBLAFMAVQBNAD4APABMAEEAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC8AMQAyADcALgAw
AC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQBuAGEAZwBlAHIAL
gBhAHMAbQB4ADwALwBMAEEAXwBVAFIATAA+ADwATABVAEkAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC
8AMQAyADcALgAwAC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQB
uAGEAZwBlAHIALgBhAHMAbQB4ADwALwBMAFUASQBfAFUAUgBMAD4APAAvAEQAQQBUAEEAPgA8AC8A
VwBSAE0ASABFAEEARABFAFIAPgA="
            }
          }
        ]
      },
      {
        "drmName": "PLAYREADY",
        "streamingProtocol": "DASH",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "position": 1
            },
            "drmData": {
              "drmSystemId": "9a04f079-9840-4286-ab92-e65be0885f95",
              "pssh": {
                "data":
"AgMAAAEAAQD4AjwAVwBSAE0ASABFAEEARABFAFIAIAB4AG0AbABuAHMAPQAiAGgAdAB0AHAAOgAv
AC8AcwBjAGgAZQBtAGEAcwAuAG0AaQBjAHIAbwBzAG8AZgB0AC4AYwBvAG0ALwBEAFIATQAvADIAM
AAwADcALwAwADMALwBQAGwAYQB5AFIAZQBhAGQAeQBIAGUAYQBkAGUAcgAiACAAdgBlAHIAcwBpAG
8AbgA9ACIANAAuADAALgAwAC4AMAAiAD4APABEAEEAVABBAD4APABQAFIATwBUAEUAQwBUAEkATgB
GAE8APgA8AEsARQBZAEwARQBOAD4AMQA2ADwALwBLAEUAWQBMAEUATgA+ADwAQQBMAEcASQBEAD4A
QQBFAFMAQwBUAFIAPAAvAEEATABHAEkARAA+ADwALwBQAFIATwBUAEUAQwBUAEkATgBGAE8APgA8A
EsASQBEAD4ASQBCAEoAdAAvADkAZABsAE8AWQAyAEEAVABOADYAMAAzAEsATwBOAGQAZwA9AD0APA
AvAEsASQBEAD4APABDAEgARQBDAEsAUwBVAE0APgA0AFcAbQA4AHgAVwA4AHEARQBrAEEAPQA8AC8
AQwBIAEUAQwBLAFMAVQBNAD4APABMAEEAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC8AMQAyADcALgAw
AC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQBuAGEAZwBlAHIAL
gBhAHMAbQB4ADwALwBMAEEAXwBVAFIATAA+ADwATABVAEkAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC
8AMQAyADcALgAwAC4AMAAuADEALwBQAGwAYQB5AFIAZQBhAGQAeQAvAHIAaQBnAGgAdABzAG0AYQB
uAGEAZwBlAHIALgBhAHMAbQB4ADwALwBMAFUASQBfAFUAUgBMAD4APAAvAEQAQQBUAEEAPgA8AC8A
VwBSAE0ASABFAEEARABFAFIAPgA="
              }
            }
          }
        ]
      },
      {
        "drmName": "WIDEVINE",
        "streamingProtocol": "DASH",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "position": 1
            },
            "drmData": {
              "drmSystemId": "edef8ba9-79d6-4ace-a3c8-27dcd51d21ed",
              "pssh": {
                "data":
"CAESEP9tEiBl1405gEzetNyjjXYaCnZlcmltYXRyaXgiCnI9dGVzdCZzPTEqBVNEX0hE"
              }
            }
          }
        ]
      },
      {
        "drmName": "FAIRPLAY_STREAMING",
        "streamingProtocol": "HLS",
        "keyData": [
          {
            "assetKey": {
              "keyId": "ff6d122065d78d39804cdeb4dca38d76",
              "key": "8ad61564bb7dcd426293df4c17cb91a8",
              "iv": "ee7166d46ddda0af473a4a1ad6e4463c",
              "position": 1
            },
            "drmData": {
              "licenseServerUrl": "http://sp-
mr433.dev.ussd.verimatrix.com:8064/fpsa/v1.0",
              "keyUri": "r=test&s=1"
            }
          }
        ]
      }
    ]
  }
}
```

Response Parameters

HTTP Code	Type	Occurs	Description
200	KeyResponse (on page 42)	1..1	Key response object containing information about the content keys for different DRM types.

For information on other possible HTTP responses, refer to Error Codes (on page 47).

Heartbeat

Retrieves the CEI heartbeat to check if the CEI endpoint is alive.

Request Syntax

```
GET /cei/2.0/heartbeat
```

Request Parameters

None

Request Example

```
curl -v -X GET 'http://localhost:8058/cei/v2.0/heartbeat'
```

Response Example

```
HTTP/1.1 200 OK
SUCCESS
```

Response Parameters

HTTP CODE	TYPE	OCCURS	DESCRIPTION
200	HeartbeatResponse (on page 46)	1..1	Data structure that contains the heartbeat response.

For information on other possible HTTP responses, refer to Error Codes (on page 47).

Chapter 3

Available Methods: CEI v1.1

This section describes the methods available in the CEI v1.1 API.

Create Content

Generates a content item based on the passed-in parameters. When the Verimatrix VCAS system receives the Create Content request, it ensures that the content item exists in the Verimatrix VCAS database. If the content item doesn't exist, it is created.

An encoder must send this request before sending any Create/Retrieve Key request for a given content item.

Request Syntax

```
PUT /cei/{version}/content/{contentId}?siteId={siteId}
```

Request Example

```
curl -X PUT --header 'Content-Type: application/json' --header 'Accept:
application/json'
'http://localhost:8058/cei/v1.1/content/sample_video?siteId=121' -d '{
  "contentRequest": {
    "siteId": "121",
    "contentId": "sample_video",
    "contentType": "VOD",
    "streamingProtocol": "SS",
    "encryptionAlgorithm": "AES_CTR",
    "keyRotation": {
    "chainedLicense": "FALSE",
    "cryptoPeriod": "1800"
    }
  }
}'
```

Request Parameters

NAME	TYPE	OCCURS	DESCRIPTION
`version`	String	1..1	Version of the endpoint to access. Example: v1.1
`contentId`	String	1..1	Unique identifier for the content. Example: sample_video Default: none
`siteId`	Integer	0..1	Unique identifier for the site. Use a site identifier to differentiate between operators. If you do not provide a `siteId`, the method uses the local `siteId` from the license.
`body`	ContentRequest (on page 38)	1..1	Parameters to use to generate the content item.

Response Example

```
{
  "contentResponse": {
  "statusCode": "OPERATION_SUCCESS",
  "statusMessage": "Successfully created content for contentId=sample_video"
  }
}
```

Response Parameters

HTTP CODE	TYPE	OCCURS	DESCRIPTION
200	ContentResponse (on page 40)	1..1	Successful operation

Create/Retrieve Key

Creates and returns the key for the passed-in content ID. The CEI currently supports single key creation and retrieval requests only.

NOTE The attributes `keyId` and `key` parameters inside the `ContentKey` node are optional request parameters and are reserved for future use; the encoder only sets the `position` sub-element.

Request Syntax

```
PUT
/cei/{version}/content/{contentId}/position/{position}/key?siteId={siteId}
```

Request Example

```
Create/Retrieve key request example :
curl -X PUT --header 'Content-Type: application/json' --header 'Accept:
application/json'
'http://localhost:8058/cei/v1.1/content/test_get_key_01/position/1/key?siteId
=1' -d '
{
"keyRequest": {
  "contentInfo": {
    "contentId": "test_get_key_01",
    "siteId": 1,
    "keyCount": "1",
    "contentKeyList": [
      {
        "position": "1"
      }
    ]
   }
  }
}'
```

Request Parameters

NAME	TYPE	OCCURS	DESCRIPTION
`version`	String	1..1	Version of the endpoint to access. Example: v1.1
`contentId`	String	1..1	Unique identifier for the content item. Example: sample_video
`position`	Long	1..1	Epoch position. Example: 1458172978
`siteId`	Integer	0..1	Unique identifier for the site. Use a site identifier to differentiate between operators. If you do not provide a `siteId`, the method uses the local `siteId` from the license. Example: 121
`body`	KeyRequest (on page 41)	1:1	Key request details.

Response Example

```
{
"keyResponse": {
          "statusCode": "OPERATION_SUCCESS",
          "statusMessage": "Successfully retrieved key and created DRM data
for contentId = test_get_key_01 ",
          "drmList": [{
                 "drmName": "PLAYREADY",
                 "streamingProtocol": "SS",
                 "keyData": [{
                        "contentKey": {
                               "keyId": "ca645ceda247eadd417bee8fc24b5fc1",
                               "key": "e8181d31b4cfe14ade56a26c31924556",
                               "position": 1
                        },
                        "drmData": {
                               "manifestHeader":
"yAIAAAEAAQC+AjwAVwBSAE0ASABFAEEARABFAFIAIAB2AGUAcgBzAGkAbwBuAD0AIgA0AC4AMQAu
ADAALgAwACIAIAB4AG0AbABuAHMAPQAiAGgAdAB0AHAAOgAvAC8AcwBjAGgAZQBtAGEAcwAuAG0Aa
QBjAHIAbwBzAG8AZgB0AC4AYwBvAG0ALwBEAFIATQAvADIAMAAwADcALwAwADMALwBQAGwAYQB5AF
IAZQBhAGQAeQBIAGUAYQBkAGUAcgAiAD4APABEAEEAVABBAD4APABQAFIATwBUAEUAQwBUAEkATgB
GAE8APgA8AEsASQBEACAAdgBhAGwAdQBlAD0AIgA3AFYAeABrAHkAawBlAGkAMwBlAHAAQgBlACsA
NgBQAHcAawB0AGYAdwBRAD0APQAiACAAQQBMAEcASQBEAD0AIgBBAEUAUwBfAEMAVABSACIAIABDA
EgARQBDAEsAUwBVAE0APQAiAG4ANQA5AHUANwBPAFEARgBFAGMAOAA9ACIALwA +
ADwALwBQAFIATwBUAEUAQwBUAEkATgBGAE8APgA8AEwAQQBfAFUAUgBMAD4AaAB0AHQAcAA6AC8AL
wAxADAALgA0AC4ANgAuADgANAAvAFAAbABhAHkAUgBlAGEAZAB5AC8AcgBpAGcAaAB0AHMAbQBhAG
4AYQBnAGUAcgAuAGEAcwBtAHgAPAAvAEwAQQBfAFUAUgBMAD4APABMAFUASQBfAFUAUgBMAD4AaAB
0AHQAcAA6AC8ALwAxADAALgA0AC4ANgAuADgANAAvAFAAbABhAHkAUgBlAGEAZAB5AC8AcgBpAGcA
aAB0AHMAbQBhAG4AYQBnAGUAcgAuAGEAcwBtAHgAPAAvAEwAVQBJAF8AVQBSAEwAPgA8AC8ARABBA
FQAQQA + ADwALwBXAFIATQBIAEUAQQBEAEUAUgA + AA == "
                        }
                 }]
          }, {
                 "drmName": "PLAYREADY",
                 "streamingProtocol": "DASH",
                 "keyData": [{
                        "contentKey": {
                               "keyId": "ca645ceda247eadd417bee8fc24b5fc1",
                               "key": "e8181d31b4cfe14ade56a26c31924556",
                               "position": 1
                        },
                       "drmData": {
                              "drmSystemId": "9a04f079-9840-4286-ab92-
e65be0885f95",
                              "pssh": {
                                     "data":
"yAIAAAEAAQC+AjwAVwBSAE0ASABFAEEARABFAFIAIAB2AGUAcgBzAGkAbwBuAD0AIgA0AC4AMQAu
ADAALgAwACIAIAB4AG0AbABuAHMAPQAiAGgAdAB0AHAAOgAvAC8AcwBjAGgAZQBtAGEAcwAuAG0Aa
QBjAHIAbwBzAG8AZgB0AC4AYwBvAG0ALwBEAFIATQAvADIAMAAwADcALwAwADMALwBQAGwAYQB5AF
IAZQBhAGQAeQBIAGUAYQBkAGUAcgAiAD4APABEAEEAVABBAD4APABQAFIATwBUAEUAQwBUAEkATgB
GAE8APgA8AEsASQBEACAAdgBhAGwAdQBlAD0AIgA3AFYAeABrAHkAawBlAGkAMwBlAHAAQgBlACsA
NgBQAHcAawB0AGYAdwBRAD0APQAiACAAQQBMAEcASQBEAD0AIgBBAEUAUwBfAEMAVABSACIAIABDA
EgARQBDAEsAUwBVAE0APQAiAG4ANQA5AHUANwBPAFEARgBFAGMAOAA9ACIALwA +
ADwALwBQAFIATwBUAEUAQwBUAEkATgBGAE8APgA8AEwAQQBfAFUAUgBMAD4AaAB0AHQAcAA6AC8AL
wAxADAALgA0AC4ANgAuADgANAAvAFAAbABhAHkAUgBlAGEAZAB5AC8AcgBpAGcAaAB0AHMAbQBhAG
4AYQBnAGUAcgAuAGEAcwBtAHgAPAAvAEwAQQBfAFUAUgBMAD4APABMAFUASQBfAFUAUgBMAD4AaAB
0AHQAcAA6AC8ALwAxADAALgA0AC4ANgAuADgANAAvAFAAbABhAHkAUgBlAGEAZAB5AC8AcgBpAGcA
aAB0AHMAbQBhAG4AYQBnAGUAcgAuAGEAcwBtAHgAPAAvAEwAVQBJAF8AVQBSAEwAPgA8AC8ARABBA
FQAQQA + ADwALwBXAFIATQBIAEUAQQBEAEUAUgA + AA == "
                              }
                       }
                 }]
          },{
        "drmName": "WIDEVINE",
        "streamingProtocol": "DASH",
        "keyData": [
          {
            "contentKey": {
              "keyId": "66cdf06431320f4a51228eb1ae763bcf",
              "key": "e8181d31b4cfe14ade56a26c31924556",
              "position": 1
            },
            "drmData": {
              "drmSystemId": "edef8ba9-79d6-4ace-a3c8-27dcd51d21ed",
              "pssh": {
                "data":
"CAESEGbN8GQxMg9KUSKOsa52O88aClZFUklNQVRSSVgiCzEyMSZUZXN0MTIzKgVTRF9IRA=="
              }
            }
          }
        ]
      },{
```

```
                 "drmName": "ADOBE_PRIMETIME",
                 "streamingProtocol": "HLS",
                 "keyData": [{
                        "contentKey": {
                               "keyId": "ca645ceda247eadd417bee8fc24b5fc1",
                               "key": "e8181d31b4cfe14ade56a26c31924556",
                               "position": 1
                        },
                        "drmData": {
                               "licenseServerUrl": "http://localhost:8056"
                        }
                 }]
          }, {
                 "drmName": "ADOBE_PRIMETIME",
                 "streamingProtocol": "HDS",
                 "keyData": [{
                        "contentKey": {
                               "keyId": "ca645ceda247eadd417bee8fc24b5fc1",
                               "key": "e8181d31b4cfe14ade56a26c31924556",
                               "position": 1
                        },
                        "drmData": {
                               "licenseServerUrl": "http://localhost:8056"
                        }
                 }]
          }, {
                 "drmName": "ADOBE_PRIMETIME",
                 "streamingProtocol": "DASH",
                 "keyData": [{
                        "contentKey": {
                               "keyId": "ca645ceda247eadd417bee8fc24b5fc1",
                               "key": "e8181d31b4cfe14ade56a26c31924556",
                               "position": 1
                        },
                        "drmData": {
                               "drmSystemId": "f239e769-efa3-4850-9c16-
a903c6932efb",
                               "licenseServerUrl": "http://localhost:8056"
                        }
                 }]
```

```
         }, {
               "drmName": "VERIMATRIX_ITV",
               "streamingProtocol": "DASH",
               "keyData": [{
                     "contentKey": {
                           "keyId": "ca645ceda247eadd417bee8fc24b5fc1",
                           "key": "e8181d31b4cfe14ade56a26c31924556",
                           "position": 1
                     },
                     "drmData": {
                           "drmSystemId": "9a27dd82-fde2-4725-8cbc-
4234aa06ec09",
                           "manifestHeader": "<mpd></mpd>",
                           "pssh": {
                                 "data":
"PFZlcmltYXRyaXg+PFBzc2ggdmVyc2lvbj0iMSI+PEtleVVybD48IVtDREFUQVtodHRwczovL3d3
dy5leGFtcGxllmNvbS9DQUIva2V5ZmlsZT9zPTEmcj10ZXN0X2dldF9rZXlfMDEmdD1WT0QmcD0xX
V0 + PC9LZXlVcmw + PC9Qc3NoPjwvVmVyaW1hdHJpeD4 = "
                           }
                     }
               }]
         }, {
               "drmName": "VERIMATRIX_ITV",
               "streamingProtocol": "HLS",
               "keyData": [{
                     "contentKey": {
                           "keyId": "ca645ceda247eadd417bee8fc24b5fc1",
                           "key": "e8181d31b4cfe14ade56a26c31924556",
                           "position": 1
                     },
                     "drmData": {
                           "keyUrl":
"https://www.example.com/CAB/keyfile?s=1&r=test_get_key_01&t=VOD&p=1&kc=1f03c
4b67f7811e69701fa163e988c98&kd=7a836b01c9f604f2ef374e60a3060a20"
                     }
                  }
                ]
               },
               {
                   "drmName": "FAIRPLAY_STREAMING",
                   "streamingProtocol": "HLS",
                   "keyData": [
                     {
                       "contentKey": {
                         "keyId": "a29f0caf56914d4389e03c49cef8c556",
                         "key": "3c3c3c3c3c3c3c3c3c3c3c3c3c3c3c3c",
                         "iv": "d5fbd6b82ed93e4ef98ae40931ee33b7",
                         "position": 0
                     },
                     "drmData": {
                       "licenseServerUrl": "http://127.0.0.1:8064/fpsa/v1.0",
                       "keyUri": "r=test_get_key_01&s=1"
                   }
                 }
               ]
             }
          ]
      }
}
```

Response Parameters

HTTP CODE	TYPE	OCCURS	DESCRIPTION
200	KeyResponse (on page 42)	1..1	Successful operation.

Heartbeat

Retrieves the CEI heartbeat.

Request Syntax

```
GET /cei/{version}/heartbeat
```

Request Parameters

NAME	TYPE	OCCURS	DESCRIPTION
`version`	String	1..1	Version of the endpoint to access. Example: v1.1

Response Parameters

HTTP CODE	TYPE	OCCURS	DESCRIPTION
200	HeartbeatResponse (on page 46)	1..1	Successful operation

Chapter 4

Data Structures

This section describes all of the complex data structures within the CEI API.

ContentRequest

Data structure that contains information about the content item to create.

NAME	TYPE	OCCURS	DESCRIPTION
`siteId`	Integer	0..1	Unique identifier of the site for which the content is being created. Use to differentiate operators.
`contentId`	String	1..1	Unique identifier of the content item to encrypt.
`contentType`	String	1..1	Content type. Valid values: • VOD • DTV
`streamingProtocol`	String	1..1	HTTP streaming protocol. Valid values: • HDS • DASH • HLS • SS
`encryptionAlgorithm`	String	1..1	Encryption algorithm to use. Valid values: • AES_CBC • AES_CTR
`keyRotation`	KeyRotation (on page 39)	1..1	Key rotation details such as crypto period, policy, and other key rotation information.

CreateAssetResponse

Data structure returned by the Create Asset method.

NAME	TYPE	OCCURS	DESCRIPTION
`statusCode`	String	1..1	Status of operation. Valid values: • OK • INVALID_ASSETID • INVALID_ASSET_TYPE
`statusMessage`	String	0..1	Descriptive status message.

KeyRotation

Data structure that contains the key rotation specifics.

NAME	TYPE	OCCURS	DESCRIPTION
`chainedLicense`	Boolean	1..1	Reserved for future use. Flag that indicates whether the DTV content item is chained; if so, licenses are required. This flag is required if the key rotation is controlled by the Verimatrix VCAS system. The Verimatrix VCAS system provides a license in the key response which contains the basic policies and the encryption key for the content. A chained license is a series of associated licenses that collectively specify content keys and policies for one or more media files. Each license chain includes a root license and one or more leaf licenses that are associated with and dependent on the root license. Default: FALSE (case-sensitive)

Name	Type	Occurs	Description
`cryptoPeriod`	Integer	1..1	Time period for which the key is valid. If set to zero ("0"), the key is used for the life of the content. Unit: Seconds

ContentResponse

Data structure returned by the Create Content method.

Name	Type	Occurs	Description
`statusCode`	String	1..1	Status of operation. Valid values: • OPERATION_SUCCESS • INVALID_SITEID • INVALID_CONTENTID • DUPLICATE_CONTENT • UNSUPPORTED_STREAMING_PROTOCOL • INVALID_CRYPTOPERIOD • INVALID_CONTENT_TYPE • INTERNAL_ERROR • INVALID_ENCRYPTION_ALGORITHM • OPERATION_FAILURE
`statusMessage`	String	0..1	Descriptive status message.

KeyRequest

Key information object returned for a specified content item.

NAME	TYPE	OCCURS	DESCRIPTION
`contentInfo`	ContentInfo (on page 41)	1..1	Key information for the specified content item.

ContentInfo

Data structure that contains content and key information.

NAME	TYPE	OCCURS	DESCRIPTION
`contentId`	String	1..1	Unique identifier of the content item for which keys are being requested.
`siteId`	String	0..1	Unique identifier of the site to which the content is associated. Use to differentiate between operators.
`contentKeyList`	ContentKey (on page 45)	1..1	`key`/`keyId` pair generated by the Verimatrix VCAS system.
`keyCount`	Integer	1..1	Number of keys being requested. Valid values: 1 Currently, the CEI only supports single key requests. If `keyCount` does not exist or is zero, a single key is returned.

KeyResponse

Key response object containing information about the content keys for different DRM types.

Name	Type	Occurs	Description
statusCode	String	1..1	Status of operation. Valid values: • OPERATION_SUCCESS • INVALID_DRM_CONFIGURATION • INTERNAL_ERROR • INVALID_CONTENTID • INVALID_SITEID • INVALID_POSITION • OPERATION_FAILURE
statusMessage	String	0..1	Descriptive status message.
drmList	Array of DrmContent (on page 43)	1..1	Array of DRM data requested by the encoder; including keys, PSSH, and DRM information.

DrmContent

Data structure that contains the DRM information requested by the encoder.

NAME	TYPE	OCCURS	DESCRIPTION
keyData	Array of KeyData (on page 44)	1..1	Array of DRM data and content key information.
streamingProtocol	String	1..1	Type of HTTP streaming protocol. Valid values: • HLS • ADS • DASH • SS
drmName	String	1..1	Name of the DRM for which the information is being returned. Valid values: • PLAYREADY • WIDEVINE • ADOBE_PRIMETIME • VERIMATRIX_ITV • FAIRPLAY_STREAMING

KeyData

Data structure that contains the DRM data and content key information.

NAME	TYPE	OCCURS	DESCRIPTION
`contentKey`	ContentKey (on page 45)	1..1	`key`/`keyId` pair generated by the Verimatrix VCAS system.
`drmData`	DrmData (on page 45)	1..1	DRM data, such as DRM system ID, DRM name, PSSH, and manifest header.

AssetKey

CEI v2.0 and above only. Data structure that contains the `key`/`keyId` pair generated by the Verimatrix VCAS system.

NAME	TYPE	OCCURS	DESCRIPTION
`keyId`	String	0..1	Unique identifier of the content key, provided by the Verimatrix VCAS system as a normalized UUID.
`key`	String	0..1	16-byte content key used for encryption, provided by the Verimatrix VCAS system.
`position`	Long	1..1	Epoch time or index for which the encoder is requesting the key. Valid values: 0 or positive integer
`iv`	String	0..1	FairPlay Streaming only. 16-byte initialization vector associated with the content key.

ContentKey

CEI v1.1 and earlier only. Data structure that contains the `key`/`keyId` pair generated by the Verimatrix VCAS system.

NAME	TYPE	OCCURS	DESCRIPTION
`keyId`	String	0..1	Unique identifier of the content key, provided by the Verimatrix VCAS system as a normalized UUID.
`key`	String	0..1	16-byte content key used for encryption, provided by the Verimatrix VCAS system.
`position`	Long	1..1	Time for which the encoder is requesting the key. For Widevine content, this is the time since the epoch for the first key period divided by the period duration. Unit: Seconds
`iv`	String	0..1	FairPlay Streaming only. 16-byte initialization vector associated with the content key.

DrmData

Data structure that contains the DRM data.

NAME	TYPE	OCCURS	DESCRIPTION
`drmSystemId`	String	1..1	Unique identifier of the DRM system.
`manifestHeader`	String	0..1	Content protection section to be inserted into the MPD.
`licenseServerUrl`	String	0..1	Adobe Primetime/FPS only. License server URL.
`pssh`	String	0..1	PSSH signalization information blob for each DRM to be inserted in the content, as a base64 string.
`keyUrl`	String	0..1	Verimatrix ITV only. Key URL.

NAME	TYPE	OCCURS	DESCRIPTION
`keyUri`	String	0..1	FairPlay Streaming only. Value that an encoder uses as the `assetId` if the `INTERPRET_ASSET_ID` parameter in the `FPSA.INI` file is set to `contentId`. Format: `r=<url encoded contentId>&s=<siteId>` • `contentId` = URL encoded content ID. • `siteId` = If Content Distribution is deployed, this value must contain the unique ID of the site that is to receive the content.

HeartbeatResponse

Data structure that contains the heartbeat response.

NAME	TYPE	OCCURS	DESCRIPTION
`statusCode`	String	1..1	Heartbeat status code. Valid values: • OPERATION_SUCCESS • OPERATION_FAILURE
`statusMesssage`	String	0..1	Message describing the status.

Appendix A

Error Codes

The following table describes the response codes that an encoder may receive while using the CEI v2.0.

HTTP STATUS	STATUS CODE	DESCRIPTION
400	INVALID_PARAM	• `assetType` is missing or invalid Ensure that `assetType` is VOD or DTV. • `assetId` is missing
400	INVALID_SITEID	`siteId` (if present in request) is invalid Ensure that value of `siteId` is not zero or negative.
400	MALFORMED_KEYID	`keyId` is not valid Ensure that `keyId` is a 16 byte canonical UUID formatted string.
400	INVALID_POSITION	`position` is missing or negative
403	CONTENT_NOT_CREATED	DTV content limit reached for Internet TV network
404	CONTENT_NOT_FOUND	Unable to retrieve content from database due to missing content
404	KEY_NOT_FOUND	Unable to retrieve key from the database due to missing key
500	CONTENT_NOT_CREATED	Failed to add content to the Multi-DRM database due to missing network
500	KEY_NOT_CREATED	Unable to create key
500	OPERATION_FAILURE	Error in preparing DRM data
500	INTERNAL_ERROR	Unexpected error in processing request