パートナーおよびカスタマーガイド

Secure Packager および Encoder Key Exchange API の仕様

Secure Packager および Encoder Key Exchange API の仕様: パートナーおよびカスタマーガイド

Table of Contents

Secure Packager and Encoder Key Exchange とは 1
全般的アーキテクチャ 1
AWS クラウドベースのアーキテクチャ 2
開始方法 3
SPEKE を初めて利用する場合 4
関連サービスと仕様 4
用語 4
顧客オンボーディング 6
DRM プラットフォームプロバイダーでオンボードになる 6
AWS のサービスと製品での SPEKE サポート 7
AWS パートナーのサービスと製品での SPEKE サポート 8
SPEKE API 仕様 9
認証 10
AWS クラウド実装の認証 10
オンプレミス製品の認証 11
SPEKE API v1 11
SPEKE API v1 - DASH-IF 仕様のカスタマイズと制約 12
SPEKE API v1 - 標準ペイロードコンポーネント 14
SPEKE API v1 - ライブワークフローメソッド呼び出しの例 16
SPEKE API v1 - VOD ワークフローメソッド呼び出しの例 20
SPEKE API v1 - コンテンツキーの暗号化 24
SPEKE API v1 - ハートビート 27
SPEKE API v1 - キー識別子のオーバーライド 28
SPEKE API v2 30
SPEKE API v2 - DASH-IF 仕様のカスタマイズと制約 31
SPEKE API v2 - 標準ペイロードコンポーネント 35
SPEKE API v2 - 暗号化契約 40
SPEKE API v2 - ライブワークフローメソッド呼び出しの例 50
SPEKE API v2 - VOD ワークフローメソッド呼び出しの例 56
SPEKE API v2 - コンテンツキーの暗号化 62
SPEKE API v2 - キー識別子のオーバーライド 65
ライセンス 67
Creative Commons Attribution-ShareAlike 4.0 International Public License 67
ドキュメント履歴 74

AWS 用語集 77
..... lxxviii

Secure Packager and Encoder Key Exchange とは

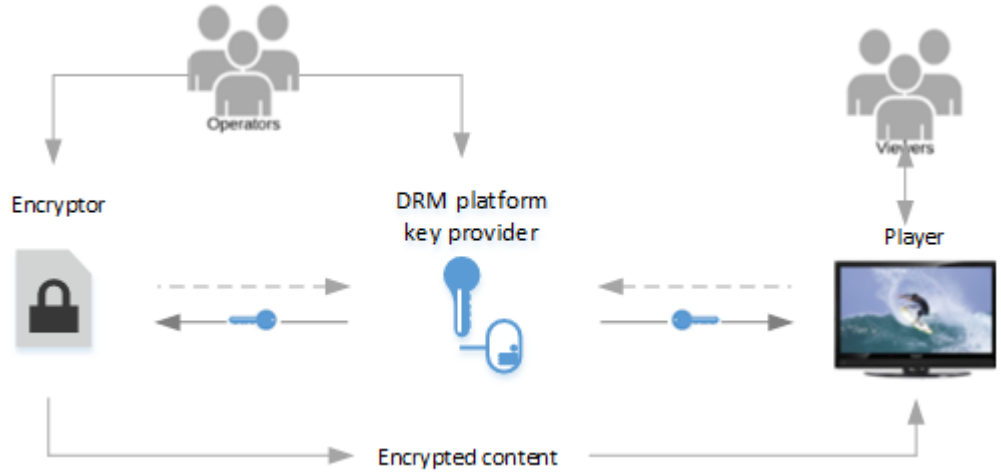
Secure Packager and Encoder Key Exchange (SPEKE) は、メディアコンテンツのエンクリプタおよびパッケージャとデジタル著作権管理 (DRM) キープロバイダー間の通信の標準を定義します。この仕様は、オンプレミスと AWS クラウドで実行されるエンクリプタに対応しています。

トピック

- 全般的アーキテクチャ
- AWS クラウドベースのアーキテクチャ
- 開始方法

全般的アーキテクチャ

次の図は、オンプレミス製品の SPEKE コンテンツ暗号化アーキテクチャの概要を示しています。

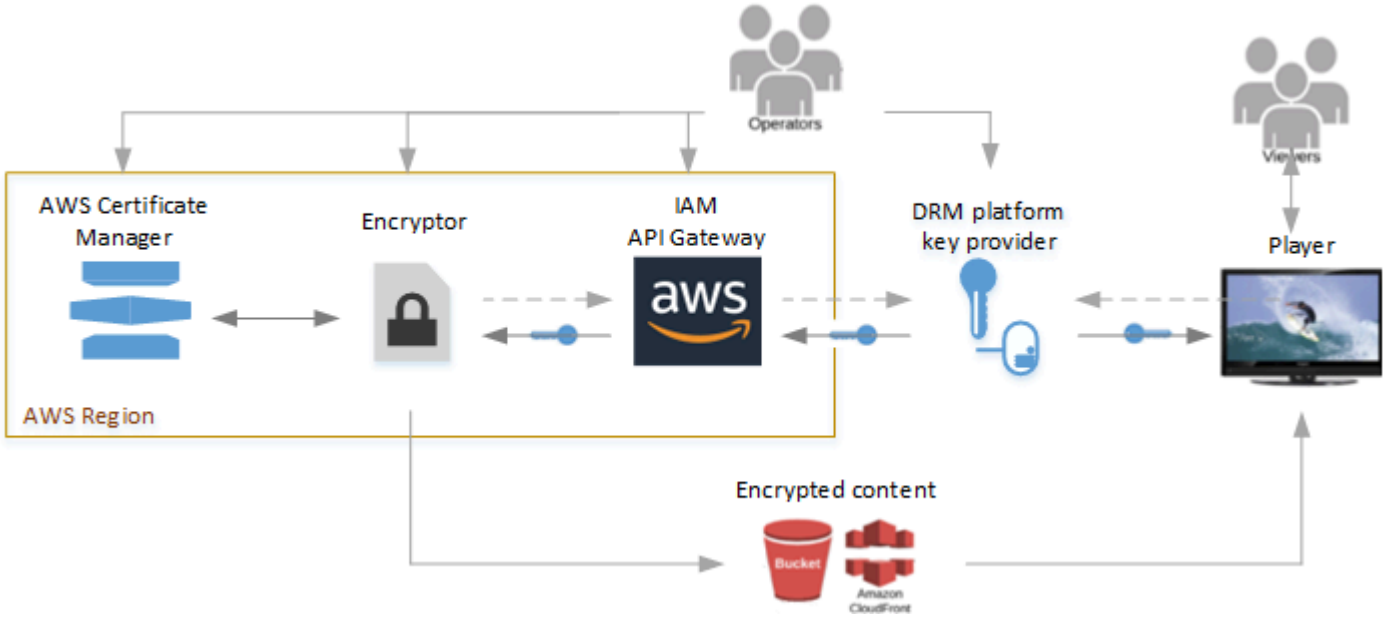


これらが、上記のアーキテクチャの主なコンポーネントです。

- エンクリプタ – 暗号化技術を提供します。オペレーターから暗号化リクエストを受け取り、DRM キープロバイダーから必要なキーを取得して、暗号化されたコンテンツを保護します。
- DRM プラットフォームキープロバイダー – SPEKE 準拠 API を介してエンクリプタに暗号化キーを提供します。メディアプレイヤーに復号化のためのライセンスも提供します。
- プレイヤー – 同じ DRM プラットフォームキープロバイダーからキーをリクエストし、取得したキーでコンテンツのロックを解除してビューワーに配信します。

AWS クラウドベースのアーキテクチャ

次の図は、AWS クラウドで実行されているサービスおよび機能で SPEKE を使用する場合のアーキテクチャの概要を示しています。



以下に主なサービスとコンポーネントを示します。

- エンクリプタ – AWS クラウドで暗号化技術を提供します。エンクリプタは、オペレーターからリクエストを受け取り、Amazon API Gateway を通じて DRM キープロバイダーから必要な暗号化キーを取得して、暗号化されたコンテンツを保護します。暗号化されたコンテンツを Amazon S3 バケットに配信するか、Amazon CloudFront ディストリビューション経由で配信します。
- AWS IAM と Amazon API Gateway – エンクリプタとキープロバイダーの間でお客様に信頼されるロールとプロキシ通信を管理します。API Gateway ではロギング機能が利用でき、お客様はエンクリプタおよび DRM プラットフォームとの関係を管理できます。お客様は、IAM ロール設定を通じてキープロバイダーアクセスを有効にします。API Gateway API は、エンクリプタと同じ AWS リージョン内に存在する必要があります。
- AWS Certificate Manager – (オプション) コンテンツキー暗号化の証明書を管理します。コンテンツキーの暗号化は、安全な通信のために推奨されるプラクティスです。証明書マネージャーは、エンクリプタと同じ AWS リージョンに存在する必要があります。
- DRM プラットフォームキープロバイダー – SPEKE 準拠 API を介してエンクリプタに暗号化キーを提供します。メディアプレイヤーに復号化のためのライセンスも提供します。
- プレイヤー – 同じ DRM プラットフォームキープロバイダーからキーをリクエストし、取得したキーでコンテンツのロックを解除してビューワーに配信します。

開始方法

SPEKE に関するその他の入門資料については、SPEKE を初めて利用する場合を参照してください。

お客様の場合

AWS Elemental DRM プラットフォームプロバイダーとのパートナーシップにより、暗号化の使用に備えてください。詳細については、顧客オンボーディングを参照してください。

DRM プラットフォームプロバイダーまたは独自のキープロバイダーを持つお客様の場合

SPEKE 仕様に従って、キープロバイダーの REST API を公開します。詳細については、SPEKE API 仕様を参照してください。

SPEKE を初めて利用する場合

このセクションでは、Secure Packager and Encoder Key Exchange (SPEKE) を初めて利用するユーザー向けに、入門資料を提供します。

SPEKE の概要については、次のウェブキャストをご覧ください。

関連サービスと仕様

- API Gateway アクセス許可 – AWS Identity and Access Management (AWS IAM) アクセス許可で API へのアクセスをコントロールする方法。
- AWS AssumeRole – AWS Security Token Service (AWS STS) を使用してロール機能を引き受ける方法。
- AWS Sigv4 – 署名バージョン 4 を使用して HTTP リクエストに署名する方法。
- DASH-IF CPIX 仕様 v2.0 – この SPEKE v1.0 仕様が基づく、DASH-IF コンテンツ保護情報交換形式 (CPIX) 仕様。
- DASH-IF CPIX 仕様 v2.3 – この SPEKE v2.0 仕様が基づく、DASH-IF コンテンツ保護情報交換形式 (CPIX) 仕様。
- DASH-IF システム ID – DRM システムの登録済みの識別子のリスト。
- https://github.com/awslabs/speke-reference-server — AWS アカウントで使用するリファレンスキープロバイダーの例。AWS での SPEKE 実装の開始に役立ちます。

用語

次のリストは、この仕様で使用される用語を定義します。可能な限り、この仕様は DASH-IF CPIX 仕様で使用されている用語に従います。

- ARN – Amazon リソースネーム。AWS リソースを一意に識別します。
- コンテンツキー – コンテンツの一部を暗号化するために使用される暗号化キー。
- コンテンツプロバイダー – 保護されたメディアを配信するためのルールと権利を提供するパブリッシャー。コンテンツプロバイダーは、ソースメディア (メザニン形式、トランスコーディング用)、アセット ID、キー識別子 (KID)、キー値、エンコーディング手順、およびコンテンツの説明メタデータを提供する場合もあります。

- DRM – デジタル著作権管理。著作権で保護されているデジタルコンテンツを未承認のアクセスから保護するために使用されます。
- DRM プラットフォーム：DRM キーやコンテンツの暗号化と復号化のためのライセンスを提供するなど、コンテンツエンクリプタとビューワーに DRM 機能とサポートを提供するシステム。
- DRM プロバイダー— DRM プラットフォームを参照してください。
- DRM システム— DRM 実装の標準。一般的な DRM システムには、Apple FairPlay、Google ワイドフォン、Microsoft などがあります PlayReady。DRM システムは、ビューワーへの配信とビューワーによるアクセスのためにデジタルコンテンツを保護する目的で、コンテンツプロバイダーによって使用されます。DASH-IF に登録されている DRM システムのリストについては、DASH-IF システム ID を参照してください。DASH-IF CPIX 仕様では、「DRM システム」という用語は、ここで定義されているとおりに使用されます。また、一部では、この仕様で DRM プラットフォームと呼ばれているものに対して「DRM システム」という用語が使用されています。
- DRM ソリューション— DRM プラットフォームを参照してください。
- DRM テクノロジー— DRM システムを参照してください。
- エンクリプタ — キープロバイダから取得したキーを使用してメディアコンテンツを暗号化するメディア処理コンポーネント。エンクリプタは通常、DRM 暗号化のシグナリングとメタデータをメディアに追加します。エンクリプタは通常、エンコーダー、パッケージャー、トランスコーダーです。
- キープロバイダー — 主要なリクエストを処理するために SPEKE REST API を公開する DRM プラットフォームのコンポーネント。キープロバイダーは、キーサーバー自体であるか、プラットフォームの別のコンポーネントである場合があります。
- キーサーバー— コンテンツの暗号化と復号化のためのキーを保持する DRM プラットフォームのコンポーネント。
- オペレーター – エンクリプタやキープロバイダーを含め、システム全体の運用の担当者。
- プレイヤー – ビューワーに代わって動作するメディアプレイヤー。メディアマニフェストファイル、メディアファイル、DRM ライセンスなど、さまざまなソースから情報を取得します。ビューワーに代わって、DRM プラットフォームからライセンスをリクエストします。

顧客オンボーディング

Secure Packager and Encoder Key Exchange (SPEKE) デジタル著作権管理 (DRM) キープロバイダーをエンクリプタおよびメディアプレイヤーと組み合わせることによって、コンテンツを不正使用から保護します。SPEKE は、メディアコンテンツのエンクリプタおよびパッケージャとデジタル著作権管理 (DRM) キープロバイダー間の通信の標準を定義します。オンボードするには、DRM プラットフォームキープロバイダーを選択し、キープロバイダーおよびエンクリプタとプレイヤー間の通信を設定します。

トピック

- DRM プラットフォームプロバイダーでオンボードになる
- AWS のサービスと製品での SPEKE サポート
- AWS パートナーのサービスと製品での SPEKE サポート

DRM プラットフォームプロバイダーでオンボードになる

以下の Amazon パートナーは、SPEKE 向けにサードパーティの DRM プラットフォーム実装を提供しています。提供サービスに関する詳細とお問い合わせ方法については、Amazon パートナーネットワークのページへのリンクをクリックしてください。リンクがないパートナーには現在 Amazon パートナーネットワークのページがありませんが、直接連絡することができます。パートナーからは、パートナーが提供するプラットフォームを使用するためのセットアップのサポートを受けられます。

DRM プラットフォームプロバイダー	SPEKE v1 サポート	SPEKE v2 サポート (AWS Elemental MediaPackage)
Axinom	√	√
BuyDRM	√	√
castLabs	√	√
EZDRM	√	√
Inisoft	√	√
INKA Entworks	√	√

DRM プラットフォームプロバイダー	SPEKE v1 サポート	SPEKE v2 サポート (AWS Elemental MediaPackage)
Insys Cloud DRM	√	√
Intertrust Technologies	√	√
Irdeto	√	√
JW Player	√	√
Kaltura	√	
NAGRA	√	√
NEXTSCAPE, Inc.	√	√
SeaChange	√	
Verimatrix	√	√
Viaccess-Orca	√	
WebStream	√	

AWS のサービスと製品での SPEKE サポート

このセクションでは、AWS クラウドで実行される AWS Media Services および AWS オンプレミスメディア製品によって提供される SPEKE サポートを示します。これらのサービスと製品は、SPEKE コンテンツ暗号化アーキテクチャのエンクリプタです。希望するストリーミングプロトコルと DRM システムが、サービスまたは製品に利用できることを確認します。

AWS サービスまたは製品	SPEKE v1 サポート	SPEKE v2 サポート	サポートされている DRM テクノロジー
AWS Elemental MediaConvert - AWS クラウドで実行されるサービス	√		ドキュメント

AWS サービスまたは製品	SPEKE v1 サポート	SPEKE v2 サポート	サポートされている DRM テクノロジー
AWS Elemental MediaPackage - AWS クラウドで実行されるサービス	√	√	ドキュメント
AWS Elemental Live - オンプレミス製品	√		ドキュメント:MPEG-DASH/HLS
AWS Elemental Server - オンプレミス製品	√		ドキュメント

AWS パートナーのサービスと製品での SPEKE サポート

このセクションでは、AWS クラウドで実行される AWS パートナーのサービスと製品によって提供される SPEKE サポートを一覧表示します。これらのサービスと製品は、SPEKE コンテンツ暗号化アーキテクチャのエンクリプタです。希望するストリーミングプロトコルと DRM システムが、サービスまたは製品に利用できることを確認します。

AWS サービスまたは製品	SPEKE v1 サポート	SPEKE v2 サポート	サポートされている DRM テクノロジー
Bitmovin Live Video Encoding	√		ドキュメント
Bitmovin Video on demand (VOD) Encoding	√		ドキュメント

SPEKE API 仕様

これは、Secure Packager and Encoder Key Exchange (SPEKE) の REST API 仕様です。この仕様を使用して、暗号化を使用するお客様に DRM 著作権保護を提供します。

動画ストリーミングワークフローでは、暗号化エンジンは DRM プラットフォームキープロバイダーと通信してコンテンツキーをリクエストします。これらのキーは非常に機密性が高いため、キープロバイダーと暗号化エンジンが安全性と信頼性の高い通信チャネルを確立することが重要です。ドキュメント内のコンテンツキーを暗号化して、より安全な end-to-end 暗号化を行うこともできます。

この仕様では、次の目標に対応しています。

- コンテンツの暗号化が必要な場合に、DRM ベンダーとお客様がエンクリプタと統合するために使用できる、シンプルで信頼性が高く、安全性の高いインターフェイスを定義します。
- VOD やライブワークフローを扱い、エンクリプタと DRM キープロバイダーのエンドポイント間において堅牢で安全性の高い通信に必要なエラー条件と認証メカニズムについて説明します。
- HLS、MSS、DASH パッケージング、および一般的な DRM システムのサポートを含めます: FairPlay PlayReady、および WideTAK/CENC。
- 今後の DRM システムをサポートするために、仕様はシンプルかつ拡張可能なものにします。
- シンプルな REST API を使用します。

> **Note**
>
>

トピック

- 認証
- SPEKE API v1

- SPEKE API v2
- ライセンス

認証

SPEKE では、オンプレミス製品と、AWS クラウドで実行されるサービスおよび機能に対して認証が必要です。

トピック

- AWS クラウド実装の認証
- オンプレミス製品の認証

AWS クラウド実装の認証

SPEKE では、エンクリプタで使用する IAM ロールによる AWS 認証が必要です。IAM ロールは、DRM プロバイダー、または AWS アカウント内の DRM エンドポイントを所有するオペレーターによって作成されます。各ロールには Amazon リソースネーム (ARN) が割り当てられ、AWS Elemental サービスオペレーターは、暗号化を要求するときにサービスコンソールで提供します。ロールのポリシーアクセス権限は、キープロバイダー API へアクセス権限を付与し、他の AWS リソースへのアクセス権限を付与しないように設定する必要があります。エンクリプタが DRM キープロバイダーに接続すると、ロール ARN を使用してキープロバイダーアカウント所有者の役割が引き継がれます。これにより、エンクリプタがキープロバイダーにアクセスするための一時的な認証情報が返されます。

一般的な実装の 1 つは、オペレーターまたは DRM プラットフォームベンダーがキープロバイダーの前で Amazon API Gateway を使用し、次に API Gateway リソースで AWS Identity and Access Management (AWS IAM) 認可を有効にすることです。次のポリシー定義の例を使用し、新しいロールにアタッチして、適切なリソースにアクセス権限を与えることができます。この場合、アクセス許可はすべての API Gateway リソースに適用されます。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "execute-api:Invoke"
            ],
```

```
            "Resource": [
                "arn:aws:execute-api:us-west-2:*:*/*/GET/*"
            ]
        }
    ]
}
```

最後に、ロールには信頼関係を追加する必要があり、オペレータはサービスを選択できる必要があります

次の例は、DRM キープロバイダーにアクセスするために作成されるロール ARN を示しています。

```
arn:aws:iam::2949266363526:role/DRMKeyServer
```

ロールの作成の詳細については、「AWS AssumeRole 」を参照してください。リクエストの署名については、「AWS Sigv4」を参照してください。

オンプレミス製品の認証

オンプレミス製品では、セキュリティを最大限に高めるために SSL/TLS とダイジェスト認証を使用することをお勧めしますが、少なくとも HTTPS 経由の基本認証を使用する必要があります。

どちらのタイプの認証でも、HTTP リクエストでは `Authorization` Clusters ヘッダーを使用します。

- ダイジェスト認証 – 認証ヘッダーは、識別子 `Digest` に続いて、リクエストを認証する一連の値で構成されます。具体的には、パスワードを安全に移動させるために使用されるサーバーからの一意の one-time-use ゼロを含む一連の MD5 ハッシュ関数を通じてレスポンス値が生成されます。
- 基本認証 – 権限ヘッダーは、識別子 `Basic` とそれに続くコロンで区切られたユーザー名とパスワードを表す base-64 でエンコードされた文字列で構成されます。

ヘッダーに関する詳細情報を含む基本およびダイジェスト認証の情報については、Internet Engineering Task Force (IETF) 仕様「RFC 2617 - HTTP 認証: 基本およびダイジェストアクセス認証」を参照してください。

SPEKE API v1

SPEKE に準拠するには、DRM キープロバイダーがこの仕様で説明されている REST API を公開する必要があります。エンクリプタはキープロバイダーへ API コールを行います。

> **Note**
>
> この仕様のコード例は、あくまでも説明用です。これらの例は完全な SPEKE 実装の一部ではないため、実行できません。

Secure Packager and Encoder Key Exchange は、DASH 業界フォーラムコンテンツ保護情報交換形式 (DASH-IF-CPIX) のデータ構造定義をキー交換に使用していますが、制限もあります。DASH-IF-CPIX は、DRM プラットフォームからエンクリプタへの拡張可能なマルチ DRM 交換を提供するスキーマを定義します これにより、コンテンツの圧縮およびパッケージング時に、すべての適応ビットレートパッケージング形式のコンテンツ暗号化が可能になります。適応ビットレートパッケージ形式には、HLS、DASH、および MSS があります。

交換形式の詳細については、https://dashif.org/docs/DASH-IF-CPIX-v2-0.pdf で DASH Industry Forum CPIX 仕様を参照してください。

トピック

- SPEKE API v1 - DASH-IF 仕様のカスタマイズと制約
- SPEKE API v1 - 標準ペイロードコンポーネント
- SPEKE API v1 - ライブワークフローメソッド呼び出しの例
- SPEKE API v1 - VOD ワークフローメソッド呼び出しの例
- SPEKE API v1 - コンテンツキーの暗号化
- SPEKE API v1 - ハートビート
- SPEKE API v1 - キー識別子のオーバーライド

SPEKE API v1 - DASH-IF 仕様のカスタマイズと制約

DASH-IF CPIX 仕様 (https://dashif.org/docs/DASH-IF-CPIX-v2-0.pdf) は、多くのユースケースとトポロジをサポートしています。SPEKE API 仕様は、次のカスタマイズと制約を伴う CPIX 仕様に準拠しています。

- SPEKE は、エンクリプタコンシューマーのワークフローに従います。
- 暗号化されたコンテンツキーの場合、SPEKE により次の制限が適用されます。
  - SPEKE は、リクエストおよびレスポンスペイロードにデジタル署名検証 (XMLDSIG) をサポートしていません。

  - SPEKE には 2048 ビットの RSA ベースの証明書が必要です。
- キーのローテーションワークフローの場合、SPEKE では ContentKeyUsageRule フィルター KeyPeriodFilter が必要です。SPEKE は他の ContentKeyUsageRule 設定をすべて無視します。
- SPEKE は UpdateHistoryItemList の機能を省略します。リストがレスポンスに存在する場合、SPEKE はそれを無視します。
- SPEKE はキーのローテーションをサポートします。SPEKE は ContentKeyPeriod`@index のみを使用してキー期間を追跡します。
- MSS をサポートするために PlayReady、SPEKE はDRMSystemタグ の下にあるカスタムパラメータを使用しますSPEKE:ProtectionHeader。
- HLS パッケージングの場合、URIExtXKey がレスポンスに存在する場合、HLS プレイリストの EXT-X-KEY タグの URI パラメータに追加するフルデータを含める必要があります。それ以上のシグナリング要件はありません。
- HLS プレイリストの場合、DRMSystem タグで、SPEKE は EXT-X-KEY タグの KEYFORMAT と KEYFORMATVERSIONS パラメータの値に対してオプションのカスタムパラメータ speke:KeyFormat と speke:KeyFormatVersions を提供します。

  HLS 初期化ベクトル (IV) は、オペレータが明示的に指定しない限り、セグメント番号の後に常に続きます。
- キーをリクエストするとき、エンクリプタは、ContentKey 要素にオプションの @explicitIV 属性を使用することがあります。キープロバイダーは、属性がリクエストに含まれていなくても、@explicitIV を使用して IV で応答することができます。
- エンクリプタはキー識別子 (KID) を生成しますが、これは与えられたコンテンツ ID とキー期間に対して同じです。キープロバイダーには、リクエストドキュメントに対するレスポンスとして KID が含まれます。
- キープロバイダーには、デバッグ目的のために自身を識別する、Speke-User-Agent レスポンスヘッダーの値を含めることができます。
- SPEKE は現在、コンテンツごとに複数のトラックやキーをサポートしていません。

  SPEKE 準拠のエンクリプタはクライアントとして機能し、POST オペレーションをキープロバイダーエンドポイントに送信します。エンクリプタは定期的に heartbeat リクエストを送信して、エンクリプタとキープロバイダーエンドポイントとの間の接続が正常であることを確認する場合があります。

SPEKE API v1 - 標準ペイロードコンポーネント

任意の SPEKE リクエストでは、1 つ以上の DRM システムのレスポンスをリクエストできます。エンクリプタは、リクエストペイロードの <cpix:DRMSystemList> で DRM システムを指定します。各システム仕様には、キーが含まれており、返されるレスポンスのタイプを示します。

次の例は、単一の DRM システム仕様を持つ DRM システムリストを示しています。

```
<cpix:DRMSystemList>
    <!--HLS AES-128 (systemId is implementation specific)-->
    <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
    systemId="81376844-f976-481e-a84e-cc25d39b0b33">
        <cpix:URIExtXKey></cpix:URIExtXKey>
        <speke:KeyFormat></speke:KeyFormat>
        <speke:KeyFormatVersions></speke:KeyFormatVersions>
    </cpix:DRMSystem>
</cpix:DRMSystemList>
```

次の表は、各 <cpix:DRMSystem> の 主要コンポーネントを示しています。

識別子	説明
systemId または schemeId	DASH IF 組織に登録されている DRM システムタイプの一意の識別子。リストについては、「DASH-IF システム ID」を参照してください。
kid	キー ID。これは実際のキーではなく、ハッシュテーブルのキーを指す識別子です。
<cpix:UriExtXKey>	標準の暗号化されていないキーをリクエストします。キーレスポンスタイプはこれか、PSSH レスポンスのいずれかである必要があります。
<cpix:PSSH>	保護システム固有ヘッダー (PSSH) をリクエストします。このタイプのヘッダーには、共通暗号化 (CENC) の一部として、DRM ベンダーの kid、systemID、およびカスタムデータへの参照が含まれています。キーレスポンスタイプはこれか、UriExtXKey レスポンスのいずれかである必要があります。

スタンダードキーと PSSH リクエストの例

次の例は、エンクリプタから DRM キープロバイダーへのサンプルリクエストの一部を示しており、主要コンポーネントが強調表示されています。最初のリクエストはスタンダードキーに対するもので、2 番目のリクエストは PSSH レスポンスに対するものです。

```
<cpix:CPIX id="abc123" xmlns:cpix="urn:dashif:org:cpix"
xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc" xmlns:speke="urn:aws:amazon:com:speke">
    <cpix:ContentKeyList>
        <cpix:ContentKey kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
        explicitIV="OFj2IjCsPJFfMAxmQxLGPw==">></cpix:ContentKey>
    </cpix:ContentKeyList>
    <cpix:DRMSystemList>
        <!-- HLS AES-128 (systemId is implementation specific)-->
        <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"  ← KID
        systemId="81376844-f976-481e-a84e-cc25d39b0b33">  ← System Id
            <cpix:URIExtXKey></cpix:URIExtXKey>  ← request Key
            <speke:KeyFormat></speke:KeyFormat>
            <speke:KeyFormatVersions></speke:KeyFormatVersions>
        </cpix:DRMSystem>

        <!-- Common encryption (Widevine)-->
        <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"  ← KID
        systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">  ← System Id
            <cpix:PSSH></cpix:PSSH>  ← request PSSH
        </cpix:DRMSystem>

    </cpix:DRMSystemList>
    ...
</cpix:CPIX>
```

スタンダードキーと PSSH のリクエストの例

次の例は、DRM キープロバイダーからエンクリプタへの対応する応答を示しています。

```
<cpix:CPIX xmlns:cpix="urn:dashif:org:cpix" xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc"
xmlns:speke="urn:aws:amazon:com:speke" id="abc123">
    <cpix:ContentKeyList>
        <cpix:ContentKey explicitIV="OFj2IjCsPJFfMAxmQxLGPw=="
        kid="98ee5596-cd3e-a20d-163a-e382420c6eff">
            <cpix:Data>
                <pskc:Secret>
                    <pskc:PlainValue>5dGAgwGuUYu4dHeHtNlxJw==</pskc:PlainValue>
                </pskc:Secret>
            </cpix:Data>
        </cpix:ContentKey>
    </cpix:ContentKeyList>
    <cpix:DRMSystemList>
        <!-- HLS AES-128 (systemId is implementation specific) -->
        <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"  <-- KID
        systemId="81376844-f976-481e-a84e-cc25d39b0b33">  <-- System Id

            <cpix:URIExtXKey>aHR0cHM6Ly83azR5dHV4cTVkLmV4ZWN1dGUtYXBpLnVzLXdlc3QtMi5hbWF6b25hd3M
            uY29tL0VrZVN0YWdlL2NsaWVudC9hYmMxMjMvOThlZTU1OTYtY2QzZS1hMjBkLTE2M2EtZTM4MjQyMGM2ZWZ
            m</cpix:URIExtXKey>  <-- Key
            <speke:KeyFormat>aWRlbnRpdHk=</speke:KeyFormat>
            <speke:KeyFormatVersions>MQ==</speke:KeyFormatVersions>
        </cpix:DRMSystem>

        <!-- Common encryption (Widevine) -->
        <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"  <-- KID
        systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">  <-- System Id

            <cpix:PSSH>AAAAanBzc2gAAAAA7e+LqXnWSs6jyCfc1R0h7QAAAEoIARIQeSIcblaNbb7Dji6sAtKZzRoNd
            2lkZXZpbmVfdGVzdCIfa2V5LWlkOmVTSWNibGFOYmI3RGppNnNBdEtaelE9PSoCU0QyAA==</cpix:PSSH>  <-- PSSH
        </cpix:DRMSystem>

    </cpix:DRMSystemList>
    ...
</cpix:CPIX>
```

SPEKE API v1 - ライブワークフローメソッド呼び出しの例

リクエストの構文例

次の URL は例であり、固定形式ではありません。

```
POST https://speke-compatible-server/speke/v1.0/copyProtection
```

リクエストボディ

CPIX 要素。

リクエストヘッダー

名前	型	発生	説明
AWS Authorization	文字列	1..1	AWS Sigv4 を参照

名前	型	発生	説明
X-Amz-Security-Token	文字列	1..1	AWS Sigv4 を参照
X-Amz-Date	文字列	1..1	AWS Sigv4 を参照
Content-Type	文字列	1..1	application/xml

レスポンスヘッダー

名前	型	発生	説明
Speke-User-Agent	文字列	1..1	キープロバイダーを識別する文字列
Content-Type	文字列	1..1	application/xml

レスポンスのリクエスト

HTTP コード	ペイロード名	発生	説明
200 (Success)	CPIX	1..1	DASH-CPIX ペイロードレスポンス
4XX (Client error)	クライアントエラーメッセージ'	1..1	クライアントエラーの説明
5XX (Server error)	サーバーエラーメッセージ	1..1	サーバーエラーの説明

> **Note**
>
> このセクションの例には、コンテンツキーの暗号化は含まれていません。コンテンツキーの暗号化を追加する方法については、コンテンツキーの暗号化を参照してください。

クリアでキーを含むリクエストペイロードのライブ例

次の例は、エンクリプタから DRM キープロバイダーへの一般的なライブリクエストペイロードを示しています。

```
<cpix:CPIX id="abc123" xmlns:cpix="urn:dashif:org:cpix"
 xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc"
 xmlns:speke="urn:aws:amazon:com:speke">
 <cpix:ContentKeyList>
  <cpix:ContentKey kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 explicitIV="OFj2IjCsPJFfMAxmQxLGPw==">></cpix:ContentKey>
 </cpix:ContentKeyList>
 <cpix:DRMSystemList>
  <!-- HLS AES-128 (systemId is implementation specific)-->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff" systemId="81376844-
f976-481e-a84e-cc25d39b0b33">
   <cpix:URIExtXKey></cpix:URIExtXKey>
   <speke:KeyFormat></speke:KeyFormat>
   <speke:KeyFormatVersions></speke:KeyFormatVersions>
  </cpix:DRMSystem>

  <!-- HLS SAMPLE-AES -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="94ce86fb-07ff-4f43-adb8-93d2fa968ca2">
   <cpix:URIExtXKey></cpix:URIExtXKey>
   <speke:KeyFormat></speke:KeyFormat>
   <speke:KeyFormatVersions></speke:KeyFormatVersions>
  </cpix:DRMSystem>

  <!-- Common encryption (Widevine)-->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
   <cpix:PSSH></cpix:PSSH>
  </cpix:DRMSystem>

  <!-- Common encryption / MSS (Playready) -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="9a04f079-9840-4286-ab92-e65be0885f95">
   <speke:ProtectionHeader></speke:ProtectionHeader>
   <cpix:PSSH></cpix:PSSH>
  </cpix:DRMSystem>
 </cpix:DRMSystemList>
 <cpix:ContentKeyPeriodList>
```

```
  <cpix:ContentKeyPeriod id="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"
 index="1" />
 </cpix:ContentKeyPeriodList>
 <cpix:ContentKeyUsageRuleList>
  <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff">
   <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f" />
  </cpix:ContentKeyUsageRule>
 </cpix:ContentKeyUsageRuleList>
</cpix:CPIX>
```

クリアでキーを含むレスポンスペイロードのライブ例

次の例は、DRM キープロバイダーからの一般的なレスポンスペイロードを示しています。

```
<cpix:CPIX xmlns:cpix="urn:dashif:org:cpix"
 xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc"
 xmlns:speke="urn:aws:amazon:com:speke" id="abc123">
 <cpix:ContentKeyList>
  <cpix:ContentKey explicitIV="OFj2IjCsPJFfMAxmQxLGPw==" kid="98ee5596-cd3e-a20d-163a-
e382420c6eff">
   <cpix:Data>
    <pskc:Secret>
     <pskc:PlainValue>5dGAgwGuUYu4dHeHtNlxJw==</pskc:PlainValue>
    </pskc:Secret>
   </cpix:Data>
  </cpix:ContentKey>
 </cpix:ContentKeyList>
 <cpix:DRMSystemList>
  <!-- HLS AES-128 (systemId is implementation specific) -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff" systemId="81376844-
f976-481e-a84e-cc25d39b0b33">

 <cpix:URIExtXKey>aHR0cHM6Ly83azR5dHV4cTVkLmV4ZWN1dGUtYXBpLnVzLXdlc3QtMi5hbWF6b25hd3MuY29tL0VrZ
cpix:URIExtXKey>
   <speke:KeyFormat>aWRlbnRpdHk=</speke:KeyFormat>
   <speke:KeyFormatVersions>MQ==</speke:KeyFormatVersions>
  </cpix:DRMSystem>

  <!-- HLS SAMPLE-AES -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="94ce86fb-07ff-4f43-adb8-93d2fa968ca2">

 <cpix:URIExtXKey>aHR0cHM6Ly83azR5dHV4cTVkLmV4ZWN1dGUtYXBpLnVzLXdlc3QtMi5hbWF6b25hd3MuY29tL0VrZ
cpix:URIExtXKey>
```

```
   <speke:KeyFormat>Y29tLmFwcGxlLnN0cmVhbWluZ2tleWRlbGl2ZXJ5</speke:KeyFormat>
   <speke:KeyFormatVersions>MQ==</speke:KeyFormatVersions>
  </cpix:DRMSystem>

  <!-- Common encryption (Widevine) -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
   <cpix:PSSH>AAAAanBzc2gAAAAA7e
+LqXnWSs6jyCfc1R0h7QAAAEoIARIQeSIcblaNbb7Dji6sAtKZzRoNd2lkZXZpbmVfdGVzdCIfa2V5LWlkOmVTSWNibGFOY
cpix:PSSH>
  </cpix:DRMSystem>

  <!-- Common encryption / MSS (Playready) -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="9a04f079-9840-4286-ab92-e65be0885f95">

 <speke:ProtectionHeader>CgMAAAEAAQAAAzwAVwBSAE0ASABFAEEARABFAFIAIAB4AG0AbABuAHMAPQAiAGgAdAB0AH
+ADwAQQBMAEcASQBEAD4AQQBFAFMAQwBUAFIAPAAvAEEATABHAEkARAA
+ADwALwBQAFIATwBUAEUAQwBUAEkATgBGAE8APgA8AEsASQBEAD4ATwBXAGoAaAB0AHIAMwB1ADkAawArAHIAZABvADEASQ
+AGgAdAB0AHAAOgAvAC8AcABsAGEAeQByAGUAYQBkAHkALgBkAGkAcgBlAGMAdAB0AGEAcABzAC4AbgBlAHQALwBwAHIALw
+ADwALwBXAFIATQBIAEUAQQBEAEUAUgA+AA==</speke:ProtectionHeader>

 <cpix:PSSH>AAADMHBzc2gAAAAAmgTweZhAQoarkuZb4IhflQAAAxAQAwAAAQABAAYDPABXAFIATQBIAEUAQQBEAEUAUgA
+ADwASwBFAFkATABFAE4APgAxADYAPAAvAEsARQBZAEwARQBOAD4APABBAEwARwBJAEQAPgBBAEUAUwBDAFQAUgA8AC8AQQ
+ADwASwBJAEQAPgBiAGgAdwBpAGUAWQAxAFcAdgBtADMARABqAGkANgBzAEEAdABLAFoAegBRAD0APQA8AC8ASwBJAEQAPg
+AGEAVABtAFAASgBWAEMAVgBaADYAcwA9ADwALwBDAEgARQBDAEsAUwBVAE0APgA8AEwAQQBfAFUAUgBMAD4AaAB0AHQAcA
+ADwALwBEAEEAVABBAD4APAAvAFcAUgBNAEgARQBBAEQARQBSAD4A</cpix:PSSH>
  </cpix:DRMSystem>
 </cpix:DRMSystemList>
 <cpix:ContentKeyPeriodList>
  <cpix:ContentKeyPeriod id="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"
 index="1" />
 </cpix:ContentKeyPeriodList>
 <cpix:ContentKeyUsageRuleList>
  <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff">
   <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f" />
  </cpix:ContentKeyUsageRule>
 </cpix:ContentKeyUsageRuleList>
</cpix:CPIX>
```

SPEKE API v1 - VOD ワークフローメソッド呼び出しの例

リクエストの構文例

次の URL は例であり、固定形式ではありません。

```
POST https://speke-compatible-server/speke/v1.0/copyProtection
```

リクエストボディ

CPIX 要素。

レスポンスヘッダー

名前	型	発生	説明
Speke-User-Agent	文字列	1..1	キープロバイダーを識別する文字列
Content-Type	文字列	1..1	application/xml

レスポンスのリクエスト

HTTP コード	ペイロード名	発生	説明
200 (Success)	CPIX	1..1	DASH-CPIX ペイロードレスポンス
4XX (Client error)	クライアントエラーメッセージ'	1..1	クライアントエラーの説明
5XX (Server error)	サーバーエラーメッセージ	1..1	サーバーエラーの説明

> **Note**
>
> このセクションの例には、コンテンツキーの暗号化は含まれていません。コンテンツキーの暗号化を追加する方法については、コンテンツキーの暗号化を参照してください。

クリアでキーを含むリクエストペイロードの VOD 例

次の例は、エンクリプタから DRM キープロバイダーへの基本的な VOD リクエストペイロードを示しています。

```
<cpix:CPIX id="abc123" xmlns:cpix="urn:dashif:org:cpix"
 xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc"
 xmlns:speke="urn:aws:amazon:com:speke">
 <cpix:ContentKeyList>
  <cpix:ContentKey kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 explicitIV="OFj2IjCsPJFfMAxmQxLGPw==">></cpix:ContentKey>
 </cpix:ContentKeyList>
 <cpix:DRMSystemList>
  <!-- HLS AES-128 (systemId is implementation specific)-->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff" systemId="81376844-
f976-481e-a84e-cc25d39b0b33">
   <cpix:URIExtXKey></cpix:URIExtXKey>
   <speke:KeyFormat></speke:KeyFormat>
   <speke:KeyFormatVersions></speke:KeyFormatVersions>
  </cpix:DRMSystem>

  <!-- HLS SAMPLE-AES -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="94ce86fb-07ff-4f43-adb8-93d2fa968ca2">
   <cpix:URIExtXKey></cpix:URIExtXKey>
   <speke:KeyFormat></speke:KeyFormat>
   <speke:KeyFormatVersions></speke:KeyFormatVersions>
  </cpix:DRMSystem>

  <!-- Common encryption (Widevine)-->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
   <cpix:PSSH></cpix:PSSH>
  </cpix:DRMSystem>

  <!-- Common encryption / MSS (Playready) -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="9a04f079-9840-4286-ab92-e65be0885f95">
   <speke:ProtectionHeader></speke:ProtectionHeader>
   <cpix:PSSH></cpix:PSSH>
  </cpix:DRMSystem>
 </cpix:DRMSystemList>
    </cpix:CPIX>
```

クリアでキーを含むレスポンスペイロードの VOD 例

次の例は、DRM キープロバイダーからの基本的な VOD レスポンスペイロードを示しています。

```
<cpix:CPIX xmlns:cpix="urn:dashif:org:cpix"
 xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc"
 xmlns:speke="urn:aws:amazon:com:speke" id="abc123">
 <cpix:ContentKeyList>
  <cpix:ContentKey explicitIV="OFj2IjCsPJFfMAxmQxLGPw==" kid="98ee5596-cd3e-a20d-163a-
e382420c6eff">
   <cpix:Data>
    <pskc:Secret>
     <pskc:PlainValue>5dGAgwGuUYu4dHeHtNlxJw==</pskc:PlainValue>
    </pskc:Secret>
   </cpix:Data>
  </cpix:ContentKey>
 </cpix:ContentKeyList>
 <cpix:DRMSystemList>
  <!-- HLS AES-128 (systemId is implementation specific) -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff" systemId="81376844-
f976-481e-a84e-cc25d39b0b33">

 <cpix:URIExtXKey>aHR0cHM6Ly83azR5dHV4cTVkLmV4ZWN1dGUtYXBpLnVzLXdlc3QtMi5hbWF6b25hd3MuY29tL0VrZ
cpix:URIExtXKey>
   <speke:KeyFormat>aWRlbnRpdHk=</speke:KeyFormat>
   <speke:KeyFormatVersions>MQ==</speke:KeyFormatVersions>
  </cpix:DRMSystem>

  <!-- HLS SAMPLE-AES -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="94ce86fb-07ff-4f43-adb8-93d2fa968ca2">

 <cpix:URIExtXKey>aHR0cHM6Ly83azR5dHV4cTVkLmV4ZWN1dGUtYXBpLnVzLXdlc3QtMi5hbWF6b25hd3MuY29tL0VrZ
cpix:URIExtXKey>
   <speke:KeyFormat>Y29tLmFwcGxlLnN0cmVhbWluZ2tleWRlbGl2ZXJ5</speke:KeyFormat>
   <speke:KeyFormatVersions>MQ==</speke:KeyFormatVersions>
  </cpix:DRMSystem>

  <!-- Common encryption (Widevine) -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
   <cpix:PSSH>AAAAanBzc2gAAAAA7e
+LqXnWSs6jyCfc1R0h7QAAAEoIARIQeSIcblaNbb7Dji6sAtKZzRoNd2lkZXZpbmVfdGVzdCIfa2V5LWlkOmVTSWNibGFOY
cpix:PSSH>
  </cpix:DRMSystem>
```

```
  <!-- Common encryption / MSS (Playready) -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="9a04f079-9840-4286-ab92-e65be0885f95">

 <speke:ProtectionHeader>CgMAAAEAAQAAAzwAVwBSAE0ASABFAEEARABFAFIAIAB4AG0AbABuAHMAPQAiAGgAdAB0AH
+ADwAQQBMAEcASQBEAD4AQQBFAFMAQwBUAFIAPAAvAEEATABHAEkARAA
+ADwALwBQAFIATwBUAEUAQwBUAEkATgBGAE8APgA8AEsASQBEAD4ATwBXAGoAaAB0AHIAMwB1ADkAawArAHIAZABvADEASQ
+AGgAdAB0AHAAOgAvAC8AcABsAGEAeQByAGUAYQBkAHkALgBkAGkAcgBlAGMAdAB0AGEAcABzAC4AbgBlAHQALwBwAHIALw
+ADwALwBXAFIATQBIAEUAQQBEAEUAUgA+AA==</speke:ProtectionHeader>

 <cpix:PSSH>AAADMHBzc2gAAAAAmgTweZhAQoarkuZb4IhflQAAAxAQAwAAAQABAAYDPABXAFIATQBIAEUAQQBEAEUAUgA
+ADwASwBFAFkATABFAE4APgAxADYAPAAvAEsARQBZAEwARQBOAD4APABBAEwARwBJAEQAPgBBAEUAUwBDAFQAUgA8AC8AQQ
+ADwASwBJAEQAPgBiAGgAdwBpAGUAWQAxAFcAdgBtADMARABqAGkANgBzAEEAdABLAFoAegBRAD0APQA8AC8ASwBJAEQAPg
+AGEAVABtAFAASgBWAEMAVgBaADYAcwA9ADwALwBDAEgARQBDAEsAUwBVAE0APgA8AEwAQQBfAFUAUgBMAD4AaAB0AHQAcA
+ADwALwBEAEEAVABBAD4APAAvAFcAUgBNAEgARQBBAEQARQBSAD4A</cpix:PSSH>
  </cpix:DRMSystem>
 </cpix:DRMSystemList>
</cpix:CPIX>
```

SPEKE API v1 - コンテンツキーの暗号化

必要に応じて SPEKE 実装にコンテンツキーの暗号化を追加できます。コンテンツキーの暗号化は、コンテンツ自体を暗号化するだけでなく、転送用のコンテンツキーを暗号化することで、完全な end-to-end 保護を保証します。キープロバイダーにこれを実装していない場合は、トランスポートレイヤーの暗号化と強力な認証をセキュリティに使用することになります。

AWS クラウドで動作するエンクリプタにコンテンツキー暗号化を使用するには、お客様は証明書を AWS Certificate Manager にインポートし、暗号化アクティビティにその結果の証明書 ARN を使用します。エンクリプタは、証明書 ARN と ACM サービスを使用して、暗号化されたコンテンツキーを DRM キープロバイダーに提供します。

制限事項

SPEKE は、DASH-IF CPIX 仕様で指定されているコンテンツキー暗号化をサポートします。ただし、次の制限があります。

- SPEKE は、リクエストおよびレスポンスペイロードにデジタル署名検証 (XMLDSIG) をサポートしていません。
- SPEKE には 2048 ビットの RSA ベースの証明書が必要です。

これらの制限は、DASH-IF 仕様のカスタマイズと制約にも記載されています。

コンテンツキーの暗号化の実装

コンテンツの暗号化キーを提供するには、DRM キープロバイダーの実装に次の項目を含めます。

- リクエストペイロードとレスポンスペイロードで要素 <cpix:DeliveryDataList> を処理します。
- レスポンスペイロードの <cpix:ContentKeyList> に暗号化された値を入力します。

これらの要素の詳細については、「DASH-IF CPIX 2.0 仕様」を参照してください。

レスポンスペイロードのコンテンツキー暗号化要素の例 <cpix:DeliveryDataList>

次の例は、追加された <cpix:DeliveryDataList> 要素を太字で示しています。

```
<?xml version="1.0" encoding="UTF-8"?>
<cpix:CPIX id="example-test-doc-encryption"
    xmlns:cpix="urn:dashif:org:cpix"
    xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc"
    xmlns:speke="urn:aws:amazon:com:speke">
    <cpix:DeliveryDataList>
        <cpix:DeliveryData id="<ORIGIN SERVER ID>">
            <cpix:DeliveryKey>
                <ds:X509Data>
                    <ds:X509Certificate><X.509 CERTIFICATE, BASE-64 ENCODED></
ds:X509Certificate>
                </ds:X509Data>
            </cpix:DeliveryKey>
        </cpix:DeliveryData>
    </cpix:DeliveryDataList>
    <cpix:ContentKeyList>
     ...
    </cpix:ContentKeyList>
</cpix:CPIX>
```

レスポンスペイロードのコンテンツキー暗号化要素の例 <cpix:DeliveryDataList>

次の例は、追加された <cpix:DeliveryDataList> 要素を太字で示しています。

```
<cpix:CPIX xmlns:cpix="urn:dashif:org:cpix"
    xmlns:enc="http://www.w3.org/2001/04/xmlenc#"
    xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc"
    xmlns:speke="urn:aws:amazon:com:speke" id="hls_test_001">
```

```
    <cpix:DeliveryDataList>
        <cpix:DeliveryData id="<ORIGIN SERVER ID>">
            <cpix:DeliveryKey>
                <ds:X509Data>
                    <ds:X509Certificate><X.509 CERTIFICATE, BASE-64 ENCODED></
ds:X509Certificate>
                </ds:X509Data>
            </cpix:DeliveryKey>
            <cpix:DocumentKey Algorithm="http://www.w3.org/2001/04/xmlenc#aes256-cbc">
                <cpix:Data>
                    <pskc:Secret>
                        <pskc:EncryptedValue>
                            <enc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/
xmlenc#rsa-oaep-mgf1p" />
                            <enc:CipherData>
                                <enc:CipherValue><RSA CIPHER VALUE></enc:CipherValue>
                            </enc:CipherData>
                        </pskc:EncryptedValue>
                        <pskc:ValueMAC>qnei/5TsfUwDu+8bhsZrLjDRDngvmnUZD2eva7SfXWw=</
pskc:ValueMAC>
                    </pskc:Secret>
                </cpix:Data>
            </cpix:DocumentKey>
            <cpix:MACMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#hmac-
sha512">
                <cpix:Key>
                    <pskc:EncryptedValue>
                        <enc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/
xmlenc#rsa-oaep-mgf1p" />
                        <enc:CipherData>
                            <enc:CipherValue><RSA CIPHER VALUE></enc:CipherValue>
                        </enc:CipherData>
                    </pskc:EncryptedValue>
                    <pskc:ValueMAC>DGqdpHUfFKxdsO9+EWrPjtdTCVfjPLwwtzEcFC/j0xY=</
pskc:ValueMAC>
                </cpix:Key>
            </cpix:MACMethod>
        </cpix:DeliveryData>
    </cpix:DeliveryDataList>
    <cpix:ContentKeyList>
     ...
    </cpix:ContentKeyList>
</cpix:CPIX>
```

レスポンスペイロードのコンテンツキー暗号化要素の例 <cpix:ContentKeyList>

次の例は、レスポンスペイロードの <cpix:ContentKeyList> 要素で暗号化されたコンテンツキーの処理を示しています。これは <pskc:EncryptedValue> 要素を使用します。

```
   <cpix:ContentKeyList>
        <cpix:ContentKey kid="682681c8-69fa-4434-9f9f-1a7f5389ec02">
            <cpix:Data>
                <pskc:Secret>
                    <pskc:EncryptedValue>
                        <enc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/
xmlenc#aes256-cbc" />
                        <enc:CipherData>
                            <enc:CipherValue>NJYebfvJ2TdMm3k6v
+rLNVYb0NoTJoTLBBdbpe8nmilEfp82SKa7MkqTn2lmQBPB</enc:CipherValue>
                        </enc:CipherData>
                    </pskc:EncryptedValue>
                    <pskc:ValueMAC>t9lW4WCebfS1GP+dh0IicMs+2+jnrAmfDa4WU6VGHc4=</
pskc:ValueMAC>
                </pskc:Secret>
            </cpix:Data>
        </cpix:ContentKey>
    </cpix:ContentKeyList>
```

これに対して、以下の例は、クリアキーとして暗号化されていないコンテンツキーを持つ同様のレスポンスペイロードを示しています。これは <pskc:PlainValue> 要素を使用します。

```
    <cpix:ContentKeyList>
        <cpix:ContentKey explicitIV="OFj2IjCsPJFfMAxmQxLGPw=="
 kid="682681c8-69fa-4434-9f9f-1a7f5389ec02">
            <cpix:Data>
                <pskc:Secret>
                    <pskc:PlainValue>5dGAgwGuUYu4dHeHtNlxJw==</pskc:PlainValue>
                </pskc:Secret>
            </cpix:Data>
        </cpix:ContentKey>
    </cpix:ContentKeyList>
```

SPEKE API v1 - ハートビート

リクエストの構文例

次の URL は例であり、固定形式ではありません。

```
GET https://speke-compatible-server/speke/v1.0/heartbeat
```

レスポンスのリクエスト

HTTP コード	ペイロード名	発生	説明
200 (Success)	statusMessage	1..1	ステータスを説明するメッセージ

SPEKE API v1 - キー識別子のオーバーライド

エンクリプタは、キーを回すたびに新しいキー識別子 (KID) を作成します。リクエストで KID を DRM キープロバイダーに渡します。ほとんどの場合、キープロバイダーは同じ KID を使用して応答しますが、レスポンスの KID には異なる値を提供できます。

以下は、KID 11111111-1111-1111-1111-111111111111 のリクエスト例です。

```
   <cpix:CPIX id="abc123" xmlns:cpix="urn:dashif:org:cpix"
xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc"
xmlns:speke="urn:aws:amazon:com:speke">
     <cpix:ContentKeyList>
      <cpix:ContentKey kid="11111111-1111-1111-1111-111111111111"></cpix:ContentKey>
     </cpix:ContentKeyList>
     <cpix:DRMSystemList>
      <!-- Common encryption (Widevine)-->
      <cpix:DRMSystem kid="11111111-1111-1111-1111-111111111111"
systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
       <cpix:PSSH />
      </cpix:DRMSystem>
     </cpix:DRMSystemList>
     <cpix:ContentKeyPeriodList>
      <cpix:ContentKeyPeriod id="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"
index="1" />
     </cpix:ContentKeyPeriodList>
     <cpix:ContentKeyUsageRuleList>
      <cpix:ContentKeyUsageRule kid="11111111-1111-1111-1111-111111111111">
       <cpix:KeyPeriodFilter
periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f" />
```

```
       </cpix:ContentKeyUsageRule>
      </cpix:ContentKeyUsageRuleList>
     </cpix:CPIX>
```

次のレスポンスは、KID を 22222222-2222-2222-2222-222222222222 にオーバーライドします。

```
    <cpix:CPIX xmlns:cpix="urn:dashif:org:cpix"
 xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc"
 xmlns:speke="urn:aws:amazon:com:speke" id="abc123">
     <cpix:ContentKeyList>
      <cpix:ContentKey explicitIV="ASgwx9pQ2/2lnDzJsUxWcQ=="
 kid="22222222-2222-2222-2222-222222222222">
       <cpix:Data>
        <pskc:Secret>
         <pskc:PlainValue>p3dWaHARtL97MpT7TE916w==</pskc:PlainValue>
        </pskc:Secret>
       </cpix:Data>
      </cpix:ContentKey>
     </cpix:ContentKeyList>
     <cpix:DRMSystemList>
      <cpix:DRMSystem kid="22222222-2222-2222-2222-222222222222"
 systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
       <cpix:PSSH>AAAAanBzc2gAAAAA7e
+LqXnWSs6jyCfc1R0h7QAAAEoIARIQeSIcblaNbb7Dji6sAtKZzRoNd2lkZXZpbmVfdGVzdCIfa2V5LWlkOmVTSWNibGFOY
cpix:PSSH>
      </cpix:DRMSystem>
     </cpix:DRMSystemList>
     <cpix:ContentKeyPeriodList>
      <cpix:ContentKeyPeriod id="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"
 index="1" />
     </cpix:ContentKeyPeriodList>
     <cpix:ContentKeyUsageRuleList>
      <cpix:ContentKeyUsageRule kid="22222222-2222-2222-2222-222222222222">
       <cpix:KeyPeriodFilter
 periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f" />
      </cpix:ContentKeyUsageRule>
     </cpix:ContentKeyUsageRuleList>
    </cpix:CPIX>
```

SPEKE API v2

SPEKE に準拠するには、DRM キープロバイダーがこの仕様で説明されている REST API を公開する必要があります。エンクリプタはキープロバイダーへ API コールを行います。

> **Note**
>
> この仕様のコード例は、あくまでも説明用です。これらの例は完全な SPEKE 実装の一部ではないため、実行できません。

Secure Packager and Encoder Key Exchange は、DASH 業界フォーラムコンテンツ保護情報交換形式 (DASH-IF-CPIX) のデータ構造定義をキー交換に使用していますが、制限もあります。DASH-IF-CPIX は、DRM プラットフォームからエンクリプタへの拡張可能なマルチ DRM 交換を提供するスキーマを定義します これにより、コンテンツの圧縮およびパッケージング時に、すべての適応ビットレートパッケージング形式のコンテンツ暗号化が可能になります。適応ビットレートパッケージ形式には、HLS、DASH、および MSS があります。

バージョン 2.0 以降、SPEKE は特定の CPIX バージョンに合わせて調整されます。

SPEKE 側では、これは `X-Speke-Version` HTTP ヘッダーを使用して適用され、CPIX 側では `CPIX@version` 属性を使用して適用されます。リクエストにこれらの要素が欠落していることは、SPEKE v1 レガシーワークフローでは一般的です。SPEKE v2 ワークフローでは、キープロバイダーが両方のバージョンパラメータをサポートしている場合にのみ CPIX ドキュメントを処理することが想定されます。

交換形式の詳細については、DASH Industry Forum の CPIX 2.3 仕様を参照してください。

大まかに、SPEKE v2.0 は SPEKE v1.0 と比較して以下の進化を遂げました。

- SPEKE XML 名前空間のすべてのタグは、CPIX XML 名前空間内の同等のタグを優先して非推奨になりました。
- `SPEKE:ProtectionHeader` は非推奨であり、`CPIX:DRMSystem.SmoothStreamingProtectionHeaderData` に置き換えられます
- `CPIX:URIExtXKey`、`SPEKE:KeyFormat`、および `SPEKE:KeyFormatVersions` は非推奨であり、`CPIX:DRMSystem.HLSSignalingData` に置き換えられます
- `CPIX@id` は `CPIX@contentId` に置き換えられます
- 新しい必須の CPIX 属性: `CPIX@version`、`ContentKey@commonEncryptionScheme`

- 新しいオプションの CPIX 要素: `DRMSystem.ContentProtectionData`
- 複数のコンテンツキーのサポート
- SPEKE と CPIX 間のクロスバージョン管理メカニズム
- HTTP ヘッダーの進化: 新しい `X-Speke-Version` ヘッダー、`Speke-User-Agent` ヘッダーが `X-Speke-User-Agent` に名前変更
- ハートビート API の非推奨

SPEKE v1.0 の仕様は変更されないため、SPEKE v1.0 ワークフローを継続してサポートするために既存の実装を変更する必要はありません。

トピック

- SPEKE API v2 - DASH-IF 仕様のカスタマイズと制約
- SPEKE API v2 - 標準ペイロードコンポーネント
- SPEKE API v2 - 暗号化契約
- SPEKE API v2 - ライブワークフローメソッド呼び出しの例
- SPEKE API v2 - VOD ワークフローメソッド呼び出しの例
- SPEKE API v2 - コンテンツキーの暗号化
- SPEKE API v2 - キー識別子のオーバーライド

SPEKE API v2 - DASH-IF 仕様のカスタマイズと制約

DASH Industry Forum CPIX 2.3 仕様は、多くのユースケースとトポロジをサポートしています。SPEKE API v2.0 仕様では、CPIX プロファイルと CPIX 用の API の両方が定義されています。この 2 つの目標を達成するために、次のカスタマイズと制約を適用した CPIX 仕様に準拠しています。

CPIX プロファイル

- SPEKE は、エンクリプタコンシューマーのワークフローに従います。
- 暗号化されたコンテンツキーの場合、SPEKE により次の制限が適用されます。
  - SPEKE は、リクエストおよびレスポンスペイロードにデジタル署名検証 (XMLDSIG) をサポートしていません。
  - SPEKE には 2048 ビットの RSA ベースの証明書が必要です。
- SPEKE は CPIX 機能のサブセットのみを利用します。

  - SPEKE は UpdateHistoryItemList の機能を省略します。リストがレスポンスに存在する場合、SPEKE はそれを無視します。
  - SPEKE では、ルート/リーフキー機能は省略されます。ContentKey@dependsOnKey 属性がレスポンスに存在する場合、SPEKE はそれを無視します。
  - SPEKE では、BitrateFilter 要素と VideoFilter@wcg 属性は省略されます。これらの要素または属性が CPIX ペイロードに存在する場合、SPEKE はそれを無視します。
- スタンダードペイロードコンポーネントのページまたは暗号化契約のページで「サポートされる」として参照されている要素または属性のみ、SPEKE v2 と交換される CPIX ドキュメントで使用できます。
- すべての要素および属性は、エンクリプタが CPIX リクエストに含めている場合、キープロバイダーの CPIX レスポンスで有効な値を保持します。そうでない場合、エンクリプタは停止してエラーをスローします。
- SPEKE は KeyPeriodFilter 要素でキーローテーションをサポートします。SPEKE は ContentKeyPeriod@index のみを使用して、キー期間を追跡します。
- HLS シグナリングの場合、複数の DRMSystem.HLSSignalingData 要素を使用する必要があります。DRMSystem.HLSSignalingData@playlist 属性値「media」で使用されるものと、DRMSystem.HLSSignalingData@playlist 属性値「master」で使用されるものです。
- キーをリクエストするとき、エンクリプタは、ContentKey 要素にオプションの @explicitIV 属性を使用することがあります。キープロバイダーは、属性がリクエストに含まれていなくても、@explicitIV を使用して IV で応答することができます。
- エンクリプタはキー識別子 (KID) を生成しますが、これは与えられたコンテンツ ID とキー期間に対して同じです。キープロバイダーには、リクエストドキュメントに対するレスポンスとして KID が含まれます。
- エンクリプタは CPIX@contentId 属性の値を含めます。この属性に空の値を受け取ると、キープロバイダーは「Missing CPIX@contentId」(CPIX@contentId がありません) という説明を含むエラーを返します。CPIX@contentId 値をキープロバイダーでオーバーライドすることはできません。

  CPIX@id 値が null でない場合は、キープロバイダーによって無視されます。
- エンクリプタは CPIX@version 属性の値を含めます。この属性に空の値を受け取ると、キープロバイダーは「Missing CPIX@version」(CPIX@version がありません) という説明を含むエラーを返します。サポートされていないバージョンのリクエストを受け取った場合、キープロバイダーから返されるエラーの説明は「Unsupported CPIX@version」(サポートされていない CPIX@version です) になります。

CPIX@version 値をキープロバイダーでオーバーライドすることはできません。

- エンクリプタは、リクエストされたキーごとに ContentKey@commonEncryptionScheme 属性の値を含めます。この属性に空の値を受け取ると、キープロバイダーは「@for ContentKeycommonEncryptionScheme KID を見逃す」というエラーを返しますid。

  一意の CPIX ドキュメントでは、異なる ContentKey@commonEncryptionScheme 属性に複数の値を混在させることはできません。このような組み合わせを受け取ると、キープロバイダーは「非準拠の ContentKey@commonEncryptionScheme combination」という説明のエラーを返すものとします。

  すべての ContentKey@commonEncryptionScheme 値がすべての DRM テクノロジと互換性があるわけではありません。このような組み合わせを受け取ると、キープロバイダーは ContentKey「@commonEncryptionScheme non compatible with DRMSystemid」という説明のエラーを返すものとします。

  ContentKey@commonEncryptionScheme 値をキープロバイダーでオーバーライドすることはできません。
- CPIX レスポンス本文で DRMSystem@PSSH と DRMSystem.ContentProtectionData innerXML <pssh> 要素に異なる値を受け取った場合、エンクリプタは停止してエラーをスローします。

CPIX 用の API

- キープロバイダーは、X-Speke-User-Agent HTTP レスポンスヘッダーの値を含めます。
- SPEKE 準拠のエンクリプタはクライアントとして機能し、POST オペレーションをキープロバイダーエンドポイントに送信します。
- エンクリプタは HTTP リクエストヘッダーの値を含め、SPEKE v2X-Speke-Version.0 の「2.0」のように MajorVersionMinorVersion、リクエストで使用される SPEKE バージョンを とします。キープロバイダーがエンクリプタによって使用される SPEKE バージョンを現在のリクエストでサポートしていない場合、キープロバイダーは「Unsupported SPEKE version」(サポート対象外の SPEKE バージョン) という説明を含むエラーを返し、ベストエフォートベースでの CPIX ドキュメントの処理を試行しません。

  キープロバイダーは、エンクリプタによって定義された X-Speke-Version ヘッダー値をリクエストへのレスポンスで変更できません。
- レスポンス本文でエラーを受信すると、エンクリプタはエラーをスローし、SPEKE v1.0 のバージョニングでリクエストを再試行しません。

キープロバイダーがエラーを返さないものの、必須情報を含む CPIX ドキュメントを返さない場合、エンクリプタは停止してエラーをスローする必要があります。

次の表は、キープロバイダーがメッセージの本文で返す必要がある標準メッセージをまとめたものです。エラーの場合の HTTP レスポンスコードは 4XX または 5XX になり、200 ではありません。422 エラーコードは SPEKE/CPIX に関連するすべてのエラーに使用できます。

エラーケース	エラーメッセージ
CPIX @contentId が定義されていない	Missing CPIX@contentId (CPIX@contentId がありません)
CPIX @version が定義されていない	Missing CPIX@version (CPIX@version がありません)
CPIX@version がサポートされていない	Unsupported CPIX@version (サポート対象外の CPIX@version)
ContentKey@ commonEncryptionScheme は定義されていません	ContentKey@commonEncryptionScheme for KID がありません `id` (は ContentKey@kid 値と`id`等しくなります)
単一の CPIX ドキュメントで使用される複数の ContentKey@commonEncryptionScheme values	非準拠の ContentKey@commonEncryption Scheme combination
ContentKey@ commonEncryptionScheme は DRM テクノロジーと互換性がありません	ContentKeyDRMSystem との互換性comm onEncryptionScheme がない `id` (は DRMSystem @systemId 値と`id`等しい)syste mId
X-Speke-Version ヘッダー値がサポートされている SPEKE バージョンではない	Unsupported SPEKE version (サポート対象外の SPEKE バージョン)
暗号化契約の形式が正しくない	Malformed encryption contract (不正な形式の暗号化契約)

エラーケース	エラーメッセージ
暗号化契約が DRM のセキュリティレベルの制約と矛盾している	Requested CPIX encryption contract not supported (リクエストされた CPIX 暗号化契約はサポートされていません)
暗号化契約に VideoFilter または AudioFilter 要素が含まれていない	Missing CPIX encryption contract (CPIX 暗号化契約がありません)

SPEKE API v2 - 標準ペイロードコンポーネント

1 回の SPEKE 要求により、エンクリプタは特定のコンテンツに対して定義された暗号化契約に従って、複数のコンテンツキーを、複数のパッケージ形式に必要なマニフェストシグナリングとともにリクエストすることができます。

これらすべての側面をカバーするために、標準の CPIX ドキュメントは、3 つの必須リストセクションと、ライブコンテンツのキーローテーションのためのオプションのリストセクションで構成されています。

<cpix:ContentKeyList> セクションおよび最上位 <cpix:CPIX> 要素

これは、ライブストリーミングと VOD ストリーミングの両方に関連した必須セクションで、エンクリプタで使用する必要があるさまざまなコンテンツキーを定義します。<cpix:ContentKeyList> 要素には 1 つまたは複数の <cpix:ContentKey> 子要素を含めることができます。それぞれの要素は個別のコンテンツキーについて記述します。

CPIX 仕様に従って、ContentKey@commonEncryptionScheme 属性に使用できる値は、ISO ベースメディアファイル形式ファイルの一般的な暗号化の仕様 (ISO/IEC 23001-7:2016) で定義されています。

- 'cenc': AES-CTR モードのフルサンプルとビデオ NAL サブサンプル暗号化
- 'cbc1': AES-CBC モードのフルサンプルとビデオ NAL サブサンプル暗号化
- 'cens': AES-CTR モードの部分的なビデオ NAL パターン暗号化
- 'cbcs': AES-CBC モードの部分的なビデオ NAL パターン暗号化

次の例は、暗号化されていない単一のコンテンツキーを持つ CPIX ドキュメントを示しています。

```
<cpix:CPIX contentId="abc123" version="2.3" xmlns:cpix="urn:dashif:org:cpix"
 xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc">
 <cpix:ContentKeyList>
  <cpix:ContentKey explicitIV="OFj2IjCsPJFfMAxmQxLGPw==" kid="98ee5596-cd3e-a20d-163a-
e382420c6eff" commonEncryptionScheme="cbcs">
   <cpix:Data>
    <pskc:Secret>
     <pskc:PlainValue>5dGAgwGuUYu4dHeHtNlxJw==</pskc:PlainValue>
    </pskc:Secret>
   </cpix:Data>
  </cpix:ContentKey>
 </cpix:ContentKeyList>
 ...
</cpix:CPIX>
```

デフォルトでは、以下の例のように、コンテンツキーは暗号化されません。ただし、<cpix:DeliveryDataList> 要素を含めることで、エンクリプタがコンテンツキーの暗号化をリクエストできます。詳細については、コンテンツキーの暗号化のセクションを参照してください。

SPEKE でサポートされる要素	必須属性	オプションの属性	必須の子要素	オプションの子要素
<cpix:CPIX>	contentId、version、xmlns:cpix、xmlns:pskc	name、xmlns:enc	1 <cpix:ContentKeyList>、1 <cpix:DRMSystemList>、1 <cpix:ContentKeyUsageRuleList>	1 <cpix:DeliveryDataList>、1 <cpix:ContentKeyPeriodList>
<cpix:ContentKeyList>	-	id	少なくとも 1 つの <cpix:ContentKey>	-
<cpix:ContentKey>	kid、commonEncryptionSchemeData	id、Algorithm、explicitIV	1 つの <pskc:Secret>	-

SPEKE でサポートされる要素	必須属性	オプションの属性	必須の子要素	オプションの子要素
<pskc:Secret>	PlainValue または EncryptedValue	ValueMAC	-	<enc:EncryptionMethod>、<enc:CipherData>

<cpix:DRM SystemList> セクション

これは、ライブストリーミングと VOD ストリーミングの両方に関連した必須セクションで、コンテンツキーとともに使用する必要があるさまざまな DRM システムを定義します。

次の例は、単一の DRM システム仕様を持つ PlayReady DRM システムリストを示しています。

```
<cpix:DRMSystemList>
 <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="9a04f079-9840-4286-ab92-e65be0885f95">
  <cpix:HLSSignalingData playlist="media">HicXmbZ2m[...]4==</cpix:HLSSignalingData>
  <cpix:HLSSignalingData playlist="master">HicXmbZ2m[...]jEi</cpix:HLSSignalingData>
  <cpix:ContentProtectionData>t7WwH24FI[...]YCC</cpix:ContentProtectionData>
  <cpix:PSSH>FFFFanBzc[...]A==</cpix:PSSH>
  <cpix:SmoothStreamingProtectionHeaderData>s5RrJ12HL[...]UBB</
cpix:SmoothStreamingProtectionHeaderData>
 </cpix:DRMSystem>
</cpix:DRMSystemList>
```

DRM SystemID の完全なリストについては、DASH-IF 識別子リポジトリのコンテンツ保護セクションを参照してください。

SPEKE でサポートされる要素	必須属性	オプションの属性	必須の子要素	オプションの子要素
<cpix:DRMSystemList>	-	id	少なくとも 1 つの <cpix:DRMSystem>	-
<cpix:DRMSystem>	kid、systemId	id、name、PSSH	-	ContentProtectionData、

SPEKE でサポートされる要素	必須属性	オプションの属性	必須の子要素	オプションの子要素
				SmoothStreamingProtectionHeaderData2 つの <cpix:HLSSignalingData> 要素と異なるプレイリスト属性値

ISO-BMFF カプセル化がメディアセグメントに適用されている場合、DRMSystem@PSSH は必須です。DRMSystem.ContentProtectionData innerXML <pssh> 要素は、マニフェストシグナリングの目的でのみ、エンクリプタによって利用されます。

DRMSystem@PSSH が存在し、DRMSystem.ContentProtectionData に innerXML <pssh> 要素が含まれている場合、両方の値は同じでなければなりません。

DRMSystem シグナリングが HLS マニフェストで伝送される場合、CPIX リクエストとレスポンスには、<cpix:HLSSignalingData playlist="media"> 要素と <cpix:HLSSignalingData playlist="master"> 要素の両方が含まれている必要があります。

<cpix:ContentKeyPeriodList> セクション

これは、ライブストリーミングにのみ関連するオプションのセクションで、コンテンツに適用される暗号化期間を定義します。

<cpix:ContentKeyPeriodList> 要素には 1 つまたは複数の <cpix:ContentKeyPeriod> 子要素を含めることができます。それぞれの要素はライブタイムラインの個別の暗号化期間について記述します。id 属性の値の一部として UUID を使用することは、一般的に使用されるアプローチです。

```
<cpix:ContentKeyPeriodList>
 <cpix:ContentKeyPeriod id="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f" index="1" /
>
</cpix:ContentKeyPeriodList>
```

SPEKE でサポートされる要素	必須属性	オプションの属性	必須の子要素	オプションの子要素
<cpix:ContentKeyPeriodList >	-	id	少なくとも 1 つの <cpix:ContentKeyPeriod >	-
<cpix:ContentKeyPeriod >	id、index	-	-	-

暗号化期間を使用する場合は、次のセクションに示すように、CPIX ドキュメント内のいずれかの暗号化期間に暗号化キーをアタッチする必要があります。

<cpix:ContentKeyUsageRuleList> セクション

これは、ライブストリーミングと VOD ストリーミングの両方に関連する必須セクションであり、暗号化期間にわたってストリームセット内でさまざまなコンテンツキーがトラックを保護する方法を定義します。

<cpix:ContentKeyUsageRuleList> 要素には、1 つまたは複数の <cpix:ContentKeyUsageRule> 子要素を含めることができます。各要素は、特定の暗号化期間中に特定のコンテンツキーがエンクリプタによって適用されるトラックを記述します。<cpix:AudioFilter> 要素に少なくとも 1 つの <cpix:VideoFilter> ContentKeyUsageRule要素が存在する必要があります。

次の例は、特定の暗号化期間中にすべてのオーディオトラックとビデオトラックに 1 つのコンテンツキーを適用するというルールのみを含む単純なリストを示しています。

```
<cpix:ContentKeyUsageRuleList>
 <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="ALL">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:AudioFilter />
  <cpix:VideoFilter />
 </cpix:ContentKeyUsageRule>
</cpix:ContentKeyUsageRuleList>
```

SPEKE でサポートされる要素	必須属性	オプションの属性	必須の子要素	オプションの子要素
<cpix:ContentKeyUsageRuleList>	-	id	少なくとも 1 つの <cpix:ContentKeyUsageRule>	-
<cpix:ContentKeyUsageRule>	kid、 intendedTrackType	-	少なくとも 1 つの <cpix:AudioFilter> または 1 つの <cpix:VideoFilter> (*)	<cpix:KeyPeriodFilter>
<cpix:KeyPeriodFilter>	periodId	-	-	-
<cpix:AudioFilter>	-	minChannels、maxChannels	-	-
<cpix:VideoFilter>	-	minPixels、maxPixels、hdr、minFps、maxFps	-	-

(*) ストリームセット内の 1 つまたは複数のトラックを保護するために単一または複数のコンテンツキーを使用することに関する詳細な説明については、ドキュメントの暗号化契約セクションを参照してください。_

SPEKE API v2 - 暗号化契約

暗号化契約は、トラックの特性に基づいて、どのコンテンツキーが特定のストリームセット内のどのトラックを保護しているかを定義します。

ストリームセット内の異なるトラックに複数のコンテンツキーを使用することが、必須ではないものの、業界のベストプラクティスとして推奨されています。少なくとも 2 つの異なるコンテンツキー (オーディオトラック用に 1 つとビデオトラック用に 1 つ) を使用するようにしてください。単一の

コンテンツキーを使用して複数のトラックを暗号化することは可能ですが、エンクリプタからキープロバイダーに送信される CPIX ドキュメントで明示的に通知する必要があります。一般的に、エンクリプタは、必要なコンテンツキーの数と、それらがさまざまなメディアトラックの暗号化にどのように活用されるかを常に正確に記述します。

原則

暗号化契約は、CPIX ドキュメントの <cpix:ContentKeyUsageRuleList> セクションにあります。このセクションでは、<cpix:ContentKeyList> セクションで定義された各コンテンツキーは、特定の <cpix:ContentKeyUsageRule> 要素に対応します。これには以下が含まれます。

- 1 つ以上のサブコンポーネントを参照できる ContentKeyUsageRule@intendedTrackType 属性。複数のサブコンポーネントが使用されている場合は、「+」記号で区切られます。ContentKeyUsageRule@intendedTrackType の値は暗号化契約では一意であり、複数の ContentKeyUsageRule 要素で使用することはできません。
- 1 つまたは複数の <cpix:AudioFilter> または <cpix:VideoFilter> 子要素 (ContentKeyUsageRule@intendedTrackType 属性の値に応じます)。

この関係を規定するルールは次のとおりです。

- ストリームセットのすべてのオーディオトラックとビデオトラックを一意のコンテンツキーで保護する必要がある場合は、文字列 'ALL' を ContentKeyUsageRule@intendedTrackType 属性値として使用する必要があります。例 1 は、このようなユースケースを示しています。この状況では、属性を持たない <cpix:AudioFilter /> 子属性と <cpix:VideoFilter /> 子属性の両方を含める必要があります。<cpix:AudioFilter> 要素や <cpix:VideoFilter> 要素のその他の組み合わせは、この特定のコンテキストでは無効です。
- それ以外のすべてのユースケースでは、ContentKeyUsageRule@intendedTrackType 属性の値は自由に定義でき、<cpix:AudioFilter /> 子要素と <cpix:VideoFilter /> 子要素の数は、「+」記号で集計されたサブコンポーネントの数に対応する必要があります。例 2/3/4/5/6/7/9/10 は、ContentKeyUsageRule@intendedTrackType 属性の値に単一のサブコンポーネントが存在する場合のこの要件を示しています。例 8 は、複数のサブコンポーネントが使用される場合を示しています。ここで、ContentKeyUsageRule@intendedTrackType="SD+HD" は異なる属性値を持つ 2 つの別個の <cpix:VideoFilter> 子要素によって記述され、ContentKeyUsageRule@intendedTrackType="HDR+HFR+UHD" は異なる属性値を持つ 3 つの別個の <cpix:VideoFilter> 子要素によって記述されています。

フィルター

CPIX は複数のフィルタリング要素と属性を定義しますが、SPEKE はそのサブセットのみをサポートしています。次の表では、これらの違いを要約しています。

CPIX フィルターのタイプ	SPEKE の全体的なサポート	SPEKE でサポートされるフィルター属性	SPEKE でサポートされないフィルター属性
<cpix:VideoFilter >	はい	minPixels、maxPixels、hdr、minFps、maxFps (オプションの属性)	wcg
<cpix:AudioFilter >	はい	minChannels、maxChannels (オプションの属性)	
<cpix:KeyPeriodFilter >	はい	periodId (必須属性)	
<cpix:BitrateFilter >	いいえ	該当なし	該当なし
<cpix:LabelFilter >	いいえ	該当なし	該当なし

の CPIX 仕様に従って VideoFilter、[minPixels , maxPixels] は両方のディメンションのすべての包含範囲であり、(minFps , maxFps] は maxFps ディメンションに対してのみ包含されます。の場合 AudioFilter、[minChannels 、maxChannels] は両方のディメンションに含まれる範囲です。

問題のある状況

暗号化契約で提供される情報が、部分的、あいまい、または誤りである状況があります。このような場合、エンクリプタとキープロバイダーが適切に動作し、コンテンツが適切に保護されていることを保証することが重要です。次の表に、このような状況での推奨される動作を示します。

状況	エンクリプタの動作	キープロバイダーの動作
ストリームセット内の 1 つまたは複数のトラックにルール	エンクリプタは、(CPIX ペイロードの外部の) 設定を確認し、関係するトラックが暗号	該当なし: キープロバイダーはストリームセット構造を把握していません。

状況	エンクリプタの動作	キープロバイダーの動作
が適用されない (以下の例 3 を参照)	化を必要としないことを確認する必要があります。想定と異なる場合、エンクリプタはエラーをスローして、処理を停止する必要があります。	
複数のルールが重複し、特定のトラックを暗号化するために複数のコンテンツキーが提案される	エンクリプタは、ドキュメントの順序で ContentKeyUsageRule 最後に正常に評価された を適用する必要があります。	該当なし: キープロバイダーはストリームセット構造を把握していません。
暗号化契約が単一の SPEKE リクエスト/レスポンスサイクルで変更される	キープロバイダーは暗号化契約を定義する役割を負わないため、エンクリプタは例外を発生させ、処理を停止します。	この状況がそもそも発生しないようにするために、キープロバイダーが SPEKE 要求の CPIX ペイロードで受信した暗号化契約を変更しないようにする必要があります。
不正な形式の暗号化契約：intendedTrackType/Filters 基数制約の例外、サポートされていないフィルターまたは属性	この場合、誤ったコンテンツ保護が行われたり、一部のトラックが保護されないままになったりする可能性が非常に高いため、エンクリプタは例外を発生させて処理を停止し、SPEKE リクエストをキープロバイダーに送信しません。	キープロバイダーは例外を発生させ、「Malformed encryption contract」(不正な形式の暗号化契約) エラーを返します。

状況	エンクリプタの動作	キープロバイダーの動作
暗号化契約の形式は正しいが、DRM のセキュリティレベルの制約に違反している。例えば、オーディオトラックと UHD ビデオトラックの両方を保護するために、単一のコンテンツキーが要求されている	この場合、誤ったコンテンツ保護が行われる可能性が非常に高いため、エンクリプタが DRM セキュリティレベルの制約を把握している場合には、例外を発生させて処理を停止し、SPEKE リクエストをキープロバイダーに送信しません。	キープロバイダーは例外を発生させ、「Requested CPIX encryption contract not supported」(リクエストされた CPIX 暗号化契約はサポートされていません) エラーを返します。
暗号化契約がない	エンクリプタは、 VideoFilter または AudioFilter 要素を含まない CPIX ドキュメントを送信しません。	キープロバイダーは例外を発生させ、「Missing CPIX encryption contract」(CPIX 暗号化契約がありません) エラーを返します。

暗号化契約の例

例 1: すべてのオーディオトラックとビデオトラックに 1 つのコンテンツキー

```
<cpix:ContentKeyUsageRuleList>
 <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="ALL">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:AudioFilter />
  <cpix:VideoFilter />
 </cpix:ContentKeyUsageRule>
</cpix:ContentKeyUsageRuleList>
```

例 2: すべてのビデオトラックに 1 つのコンテンツキー、すべてのオーディオトラックに 1 つのコンテンツキー

```
<cpix:ContentKeyUsageRuleList>
    <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="VIDEO">
        <cpix:KeyPeriodFilter
 periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
```

```
        <cpix:VideoFilter />
    </cpix:ContentKeyUsageRule>
    <cpix:ContentKeyUsageRule kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 intendedTrackType="AUDIO">
        <cpix:KeyPeriodFilter
 periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
        <cpix:AudioFilter />
    </cpix:ContentKeyUsageRule>
</cpix:ContentKeyUsageRuleList>
```

例 3: すべてのオーディオトラックと暗号化されていないビデオトラックに 1 つのコンテンツキー

```
<cpix:ContentKeyUsageRuleList>
 <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="VIDEO">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter />
 </cpix:ContentKeyUsageRule>
</cpix:ContentKeyUsageRuleList>
```

例 4: 異なるビデオトラック (SD/HD) に複数のコンテンツキー、すべてのオーディオトラックに 1 つのコンテンツキー

```
<cpix:ContentKeyUsageRuleList>
 <!-- Rule for SD video tracks (up to 1024x576) -->
 <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="SD">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter maxPixels="589824" />
 </cpix:ContentKeyUsageRule>
 <!-- Rule for HD video tracks (more than 1024x576) -->
 <cpix:ContentKeyUsageRule kid="37e3de05-9a3b-4c69-8970-63c17a95e0b7"
 intendedTrackType="HD">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter minPixels="589825" />
 </cpix:ContentKeyUsageRule>
 <!-- Rule for all audio tracks -->
 <cpix:ContentKeyUsageRule kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 intendedTrackType="AUDIO">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:AudioFilter />
 </cpix:ContentKeyUsageRule>
```

```
</cpix:ContentKeyUsageRuleList>
```

例 5: 異なるビデオトラック (SD/HD/UHD) に複数のコンテンツキー、すべてのオーディオトラックに 1 つのコンテンツキー

```
<cpix:ContentKeyUsageRuleList>
 <!-- Rule for SD video tracks (up to 1024x576) -->
 <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="SD">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter maxPixels="589824" />
 </cpix:ContentKeyUsageRule>
 <!-- Rule for HD video tracks (more than 1024x576, up to 1920x1080) -->
 <cpix:ContentKeyUsageRule kid="37e3de05-9a3b-4c69-8970-63c17a95e0b7"
 intendedTrackType="HD">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter minPixels="589825" maxPixels="2073600" />
 </cpix:ContentKeyUsageRule>
 <!-- Rule for UHD video tracks (more than 1920x1080) -->
 <cpix:ContentKeyUsageRule kid="75c6fa78-8b5d-6d75-9653-26f41b78d1a3"
 intendedTrackType="UHD">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter minPixels="2073601" />
 </cpix:ContentKeyUsageRule>
 <!-- Rule for all audio tracks -->
 <cpix:ContentKeyUsageRule kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 intendedTrackType="AUDIO">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:AudioFilter />
 </cpix:ContentKeyUsageRule>
</cpix:ContentKeyUsageRuleList>
```

例 6: 異なるビデオトラック (SD/HD/UHD1/UHD2) に複数のコンテンツキー、すべてのオーディオトラックに 1 つのコンテンツキー

```
<cpix:ContentKeyUsageRuleList>
 <!-- Rule for SD video tracks (up to 1024x576) -->
 <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="SD">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter maxPixels="589824" />
 </cpix:ContentKeyUsageRule>
```

```
 <!-- Rule for HD video tracks (more than 1024x576, up to 1920x1080) -->
 <cpix:ContentKeyUsageRule kid="37e3de05-9a3b-4c69-8970-63c17a95e0b7"
 intendedTrackType="HD">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter minPixels="589825" maxPixels="2073600" />
 </cpix:ContentKeyUsageRule>
 <!-- Rule for UHD1 video tracks (more than 1920x1080, up to 4096x2160) -->
 <cpix:ContentKeyUsageRule kid="75c6fa78-8b5d-6d75-9653-26f41b78d1a3"
 intendedTrackType="UHD1">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter minPixels="2073601" maxPixels="8847360" />
 </cpix:ContentKeyUsageRule>
 <!-- Rule for UHD2 video tracks (more than 4096x2160) -->
 <cpix:ContentKeyUsageRule kid="63d2ec36-6b7c-9f34-4546-97d01f36f7c5"
 intendedTrackType="UHD2">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter minPixels="8847361" />
 </cpix:ContentKeyUsageRule>
 <!-- Rule for all audio tracks -->
 <cpix:ContentKeyUsageRule kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 intendedTrackType="AUDIO">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:AudioFilter />
 </cpix:ContentKeyUsageRule>
</cpix:ContentKeyUsageRuleList>
```

例 7: 異なるビデオトラック (SD/HD1/HD2/UHD1/UHD2) に複数のコンテンツキー、すべてのオーディオトラックに 1 つのコンテンツキー

```
<cpix:ContentKeyUsageRuleList>
 <!-- Rule for SD video tracks (up to 1024x576) -->
 <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="SD">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter maxPixels="589824" />
 </cpix:ContentKeyUsageRule>
 <!-- Rule for HD1 video tracks (more than 1024x576, up to 1280x720) -->
 <cpix:ContentKeyUsageRule kid="37e3de05-9a3b-4c69-8970-63c17a95e0b7"
 intendedTrackType="HD1">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter minPixels="589825" maxPixels="921600" />
 </cpix:ContentKeyUsageRule>
        <!-- Rule for HD2 video tracks (more than 1280x720, up to 1920x1080) -->
```

```
          <cpix:ContentKeyUsageRule kid="cda406d8-9d87-4f76-92da-31110e756176"
 intendedTrackType="HD2">
            <cpix:KeyPeriodFilter
 periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
            <cpix:VideoFilter minPixels="921601" maxPixels="2073600" />
          </cpix:ContentKeyUsageRule>
 <!-- Rule for UHD1 video tracks (more than 1920x1080, up to 4096x2160) -->
 <cpix:ContentKeyUsageRule kid="75c6fa78-8b5d-6d75-9653-26f41b78d1a3"
 intendedTrackType="UHD1">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter minPixels="2073601" maxPixels="8847360" />
 </cpix:ContentKeyUsageRule>
 <!-- Rule for UHD2 video tracks (more than 4096x2160) -->
 <cpix:ContentKeyUsageRule kid="63d2ec36-6b7c-9f34-4546-97d01f36f7c5"
 intendedTrackType="UHD2">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter minPixels="8847361" />
 </cpix:ContentKeyUsageRule>
 <!-- Rule for all audio tracks -->
 <cpix:ContentKeyUsageRule kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 intendedTrackType="AUDIO">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:AudioFilter />
 </cpix:ContentKeyUsageRule>
</cpix:ContentKeyUsageRuleList>
```

例 8: 異なるビデオトラック (複数の属性タイプに基づく) に複数のコンテンツキー、すべてのオーディオトラックに 1 つのコンテンツキー

```
<cpix:ContentKeyUsageRuleList>
 <!-- Rule for SD and HD video tracks-->
 <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="SD+HD">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter maxPixels="442368" maxFps="30" hdr="false"/>
  <cpix:VideoFilter minPixels="442369" maxPixels="2073600" maxFps="30" hdr="false"/>
 </cpix:ContentKeyUsageRule>
 <!-- Rule for HDR, HFR and UHD video tracks-->
 <cpix:ContentKeyUsageRule kid="37e3de05-9a3b-4c69-8970-63c17a95e0b7"
 intendedTrackType="HDR+HFR+UHD">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter hdr="true" />
  <cpix:VideoFilter minFps="30" />
```

```
  <cpix:VideoFilter minPixels="20736001" />
 </cpix:ContentKeyUsageRule>
 <!-- Rule for all audio tracks-->
 <cpix:ContentKeyUsageRule kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 intendedTrackType="AUDIO">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:AudioFilter />
 </cpix:ContentKeyUsageRule>
</cpix:ContentKeyUsageRuleList>
```

例 9: すべてのビデオトラックに 1 つのコンテンツキー、ステレオおよびマルチチャネルオーディオトラックに複数のコンテンツキー

```
<cpix:ContentKeyUsageRuleList>
 <!-- Rule for video tracks-->
 <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="VIDEO">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:VideoFilter />
 </cpix:ContentKeyUsageRule>
 <!-- Rule for stereo audio tracks-->
 <cpix:ContentKeyUsageRule kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 intendedTrackType="STEREO_AUDIO">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:AudioFilter maxChannels="2"/>
 </cpix:ContentKeyUsageRule>
 <!-- Rule for multichannel audio tracks-->
 <cpix:ContentKeyUsageRule kid="7ae8e96f-309e-42c3-a510-24023d923373"
 intendedTrackType="MULTICHANNEL_AUDIO">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <AudioFilter minChannels="3"/>
 </cpix:ContentKeyUsageRule>
</cpix:ContentKeyUsageRuleList>
```

例 10: すべてのビデオトラックに 1 つのコンテンツキー、ステレオおよび 2 つの種類のマルチチャネルオーディオトラックに複数のコンテンツキー

```
<cpix:ContentKeyUsageRuleList>
 <!-- Rule for video tracks-->
 <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="VIDEO">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
```

```
  <cpix:VideoFilter />
 </cpix:ContentKeyUsageRule>
 <!-- Rule for stereo audio tracks-->
 <cpix:ContentKeyUsageRule kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 intendedTrackType="STEREO_AUDIO">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:AudioFilter maxChannels="2"/>
 </cpix:ContentKeyUsageRule>
 <!-- Rule for multichannel audio tracks (3 to 6 channels)-->
 <cpix:ContentKeyUsageRule kid="7ae8e96f-309e-42c3-a510-24023d923373"
 intendedTrackType="MULTICHANNEL_AUDIO_3_6">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:AudioFilter minChannels="3" maxChannels="6"/>
 </cpix:ContentKeyUsageRule>
  <!-- Rule for multichannel audio tracks (7 channels and more)-->
 <cpix:ContentKeyUsageRule kid="81eb3761-55ff-4d22-a31d-94f01bbfd8ba"
 intendedTrackType="MULTICHANNEL_AUDIO_7">
  <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
  <cpix:AudioFilter minChannels="7"/>
 </cpix:ContentKeyUsageRule>
</cpix:ContentKeyUsageRuleList>
```

SPEKE API v2 - ライブワークフローメソッド呼び出しの例

リクエストの構文例

次の URL は例であり、固定形式ではありません。

```
POST https://speke-compatible-server/speke/v2.0/copyProtection
```

リクエストボディ

CPIX ドキュメント。

リクエストヘッダー

名前	型	発生	説明
AWS Authorization	文字列	1..1	AWS Sigv4 を参照

名前	型	発生	説明
X-Amz-Security-Token	文字列	1..1	AWS Sigv4 を参照
X-Amz-Date	文字列	1..1	AWS Sigv4 を参照
Content-Type	文字列	1..1	application/xml
X-Speke-Version	文字列	1..1	リクエストで使用された SPEKE API バージョン。SPEKE v2 MajorVersion.0 の場合は「2.0MinorVersion」のように として扱われます。

レスポンスヘッダー

名前	型	発生	説明
X-Speke-User-Agent	文字列	1..1	キープロバイダーを識別する文字列
Content-Type	文字列	1..1	application/xml
X-Speke-Version	文字列	1..1	リクエストで使用された SPEKE API バージョン。SPEKE v2 MajorVersion.0 の場合は「2.0MinorVersion」のように として扱われます。

レスポンスのリクエスト

HTTP コード	ペイロード名	発生	説明
200 (Success)	CPIX	1..1	DASH-CPIX ペイロードレスポンス
4XX (Client error)	クライアントエラーメッセージ'	1..1	クライアントエラーの説明
5XX (Server error)	サーバーエラーメッセージ	1..1	サーバーエラーの説明

> **Note**
>
> このセクションの例には、コンテンツキーの暗号化は含まれていません。コンテンツキーの暗号化を追加する方法については、コンテンツキーの暗号化を参照してください。

クリアでキーを含むリクエストペイロードのライブ例

次の例は、エンクリプタから DRM キープロバイダーへの一般的なライブリクエストペイロードを示しています。ここでは、すべてのビデオトラックに 1 つのコンテンツキー、すべてのオーディオトラックに 1 つのコンテンツキーがあります。

```
<cpix:CPIX contentId="abc123" version="2.3" xmlns:cpix="urn:dashif:org:cpix"
 xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc">
 <cpix:ContentKeyList>
  <cpix:ContentKey explicitIV="OFj2IjCsPJFfMAxmQxLGPw==" kid="98ee5596-cd3e-a20d-163a-
e382420c6eff" commonEncryptionScheme="cbcs"></cpix:ContentKey>
  <cpix:ContentKey explicitIV="L6jzdXrXAFbCJGBuMrrKrG==" kid="53abdba2-f210-43cb-bc90-
f18f9a890a02" commonEncryptionScheme="cbcs"></cpix:ContentKey>
 </cpix:ContentKeyList>
 <cpix:DRMSystemList>
  <!-- FairPlay -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="94ce86fb-07ff-4f43-adb8-93d2fa968ca2">
   <cpix:HLSSignalingData playlist="media"></cpix:HLSSignalingData>
   <cpix:HLSSignalingData playlist="master"></cpix:HLSSignalingData>
  </cpix:DRMSystem>
  <cpix:DRMSystem kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 systemId="94ce86fb-07ff-4f43-adb8-93d2fa968ca2">
```

```
    <cpix:HLSSignalingData playlist="media"></cpix:HLSSignalingData>
    <cpix:HLSSignalingData playlist="master"></cpix:HLSSignalingData>
   </cpix:DRMSystem>
   <!-- Widevine -->
   <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
  systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
    <cpix:HLSSignalingData playlist="media"></cpix:HLSSignalingData>
    <cpix:HLSSignalingData playlist="master"></cpix:HLSSignalingData>
    <cpix:ContentProtectionData></cpix:ContentProtectionData>
    <cpix:PSSH></cpix:PSSH>
   </cpix:DRMSystem>
   <cpix:DRMSystem kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
  systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
    <cpix:HLSSignalingData playlist="media"></cpix:HLSSignalingData>
    <cpix:HLSSignalingData playlist="master"></cpix:HLSSignalingData>
    <cpix:ContentProtectionData></cpix:ContentProtectionData>
    <cpix:PSSH></cpix:PSSH>
   </cpix:DRMSystem>
   <!-- Playready -->
   <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
  systemId="9a04f079-9840-4286-ab92-e65be0885f95">
    <cpix:HLSSignalingData playlist="media"></cpix:HLSSignalingData>
    <cpix:HLSSignalingData playlist="master"></cpix:HLSSignalingData>
    <cpix:ContentProtectionData></cpix:ContentProtectionData>
    <cpix:PSSH></cpix:PSSH>
    <cpix:SmoothStreamingProtectionHeaderData></
cpix:SmoothStreamingProtectionHeaderData>
   </cpix:DRMSystem>
   <cpix:DRMSystem kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
  systemId="9a04f079-9840-4286-ab92-e65be0885f95">
    <cpix:HLSSignalingData playlist="media"></cpix:HLSSignalingData>
    <cpix:HLSSignalingData playlist="master"></cpix:HLSSignalingData>
    <cpix:ContentProtectionData></cpix:ContentProtectionData>
    <cpix:PSSH></cpix:PSSH>
    <cpix:SmoothStreamingProtectionHeaderData></
cpix:SmoothStreamingProtectionHeaderData>
   </cpix:DRMSystem>
  </cpix:DRMSystemList>
  <cpix:ContentKeyPeriodList>
   <cpix:ContentKeyPeriod id="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"
  index="1" />
  </cpix:ContentKeyPeriodList>
  <cpix:ContentKeyUsageRuleList>
```

```
  <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="VIDEO">
   <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
   <cpix:VideoFilter />
  </cpix:ContentKeyUsageRule>
  <cpix:ContentKeyUsageRule kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 intendedTrackType="AUDIO">
   <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
   <cpix:AudioFilter />
  </cpix:ContentKeyUsageRule>
 </cpix:ContentKeyUsageRuleList>
</cpix:CPIX>
```

クリアでキーを含むレスポンスペイロードのライブ例

次の例は、DRM キープロバイダーからの一般的なレスポンスペイロードを示しています (読みやすくするために、戻り値は [...] で短縮されています)。

```
<cpix:CPIX contentId="abc123" version="2.3" xmlns:cpix="urn:dashif:org:cpix"
 xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc">
 <cpix:ContentKeyList>
  <cpix:ContentKey explicitIV="OFj2IjCsPJFfMAxmQxLGPw==" kid="98ee5596-cd3e-a20d-163a-
e382420c6eff" commonEncryptionScheme="cbcs">
   <cpix:Data>
    <pskc:Secret>
     <pskc:PlainValue>5dGAgwGuUYu4dHeHtNlxJw==</pskc:PlainValue>
    </pskc:Secret>
   </cpix:Data>
  </cpix:ContentKey>
  <cpix:ContentKey explicitIV="L6jzdXrXAFbCJGBuMrrKrG==" kid="53abdba2-f210-43cb-bc90-
f18f9a890a02" commonEncryptionScheme="cbcs">
   <cpix:Data>
    <pskc:Secret>
     <pskc:PlainValue>h3toSFIlyAYpfXVQ795m6x==</pskc:PlainValue>
    </pskc:Secret>
   </cpix:Data>
  </cpix:ContentKey>
 </cpix:ContentKeyList>
 <cpix:DRMSystemList>
  <!-- FairPlay -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="94ce86fb-07ff-4f43-adb8-93d2fa968ca2">
   <cpix:HLSSignalingData playlist="media">aHR0cHM6L[...]WZm</cpix:HLSSignalingData>
```

```
    <cpix:HLSSignalingData playlist="master">Y29tLmFwc[...]XJ5</cpix:HLSSignalingData>
   </cpix:DRMSystem>
   <cpix:DRMSystem kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
  systemId="94ce86fb-07ff-4f43-adb8-93d2fa968ca2">
    <cpix:HLSSignalingData playlist="media">trBAnbMcj[...]u44</cpix:HLSSignalingData>
    <cpix:HLSSignalingData playlist="master">mn626PjyR[...]2fi</cpix:HLSSignalingData>
   </cpix:DRMSystem>
   <!-- Widevine -->
   <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
  systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
    <cpix:HLSSignalingData playlist="media">Ifa2V5LWl[...]nNB</cpix:HLSSignalingData>
    <cpix:HLSSignalingData playlist="master">oIARIQeSI[...]Nd2l</cpix:HLSSignalingData>
    <cpix:ContentProtectionData>RoNd2lkZXZ[...]Nib</cpix:ContentProtectionData>
    <cpix:PSSH>AAAAanBzc[...]A==</cpix:PSSH>
   </cpix:DRMSystem>
   <cpix:DRMSystem kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
  systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
    <cpix:HLSSignalingData playlist="media">lTznjvtzL[...]GfJ</cpix:HLSSignalingData>
    <cpix:HLSSignalingData playlist="master">XgzdzQH7p[...]zeX</cpix:HLSSignalingData>
    <cpix:ContentProtectionData>TdgRnuJsZ[...]wDw</cpix:ContentProtectionData>
    <cpix:PSSH>mYZbjpWdS[...]D==</cpix:PSSH>
   </cpix:DRMSystem>
   <!-- Playready -->
   <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
  systemId="9a04f079-9840-4286-ab92-e65be0885f95">
    <cpix:HLSSignalingData playlist="media">HicXmbZ2m[...]4==</cpix:HLSSignalingData>
    <cpix:HLSSignalingData playlist="master">GVzdCIfa2[...]Eta</cpix:HLSSignalingData>
    <cpix:ContentProtectionData>t7WwH24FI[...]YCC</cpix:ContentProtectionData>
    <cpix:PSSH>FFFFanBzc[...]A==</cpix:PSSH>
    <cpix:SmoothStreamingProtectionHeaderData>s5RrJ12HL[...]UBB</
cpix:SmoothStreamingProtectionHeaderData>
   </cpix:DRMSystem>
   <cpix:DRMSystem kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
  systemId="9a04f079-9840-4286-ab92-e65be0885f95">
    <cpix:HLSSignalingData playlist="media">BptGzwis2[...]Iej</cpix:HLSSignalingData>
    <cpix:HLSSignalingData playlist="master">3c9SXdVa0[...]MBH</cpix:HLSSignalingData>
    <cpix:ContentProtectionData>HotJCMQyc[...]GpU</cpix:ContentProtectionData>
    <cpix:PSSH>S6UD43ybN[...]f==</cpix:PSSH>
    <cpix:SmoothStreamingProtectionHeaderData>VBFUv2or0[...]JeP</
cpix:SmoothStreamingProtectionHeaderData>
   </cpix:DRMSystem>
  </cpix:DRMSystemList>
  <cpix:ContentKeyPeriodList>
```

```
  <cpix:ContentKeyPeriod id="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"
 index="1" />
 </cpix:ContentKeyPeriodList>
 <cpix:ContentKeyUsageRuleList>
  <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="VIDEO">
   <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
   <cpix:VideoFilter />
  </cpix:ContentKeyUsageRule>
  <cpix:ContentKeyUsageRule kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 intendedTrackType="AUDIO">
   <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
   <cpix:AudioFilter />
  </cpix:ContentKeyUsageRule>
 </cpix:ContentKeyUsageRuleList>
</cpix:CPIX>
```

SPEKE API v2 - VOD ワークフローメソッド呼び出しの例

リクエストの構文例

次の URL は例であり、固定形式ではありません。

```
POST https://speke-compatible-server/speke/v2.0/copyProtection
```

リクエストボディ

CPIX ドキュメント。

リクエストヘッダー

名前	型	発生	説明
AWS Authorization	文字列	1..1	AWS Sigv4 を参照
X-Amz-Security-Token	文字列	1..1	AWS Sigv4 を参照
X-Amz-Date	文字列	1..1	AWS Sigv4 を参照
Content-Type	文字列	1..1	application/xml

名前	型	発生	説明
X-Speke-Version	文字列	1..1	リクエストで使用された SPEKE API バージョン。SPEKE v2 MajorVersion.0 の場合は「2.0MinorVersion」のように として扱われます。

レスポンスヘッダー

名前	型	発生	説明
X-Speke-User-Agent	文字列	1..1	キープロバイダーを識別する文字列
Content-Type	文字列	1..1	application/xml
X-Speke-Version	文字列	1..1	リクエストで使用された SPEKE API バージョン。SPEKE v2 MajorVersion.0 の場合は「2.0MinorVersion」のように として扱われます。

レスポンスのリクエスト

HTTP コード	ペイロード名	発生	説明
200 (Success)	CPIX	1..1	DASH-CPIX ペイロードレスポンス
4XX (Client error)	クライアントエラーメッセージ'	1..1	クライアントエラーの説明

HTTP コード	ペイロード名	発生	説明
5XX (Server error)	サーバーエラーメッセージ	1..1	サーバーエラーの説明

> **Note**
>
> このセクションの例には、コンテンツキーの暗号化は含まれていません。コンテンツキーの暗号化を追加する方法については、コンテンツキーの暗号化を参照してください。

クリアでキーを含むリクエストペイロードの VOD 例

次の例は、エンクリプタから DRM キープロバイダーへの一般的な VOD リクエストペイロードを示しています。すべてのビデオトラックに 1 つのコンテンツキー、すべてのオーディオトラックに 1 つのコンテンツキーがあります。

```
<cpix:CPIX contentId="abc123" version="2.3" xmlns:cpix="urn:dashif:org:cpix"
 xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc">
 <cpix:ContentKeyList>
  <cpix:ContentKey explicitIV="OFj2IjCsPJFfMAxmQxLGPw==" kid="98ee5596-cd3e-a20d-163a-
e382420c6eff" commonEncryptionScheme="cbcs"></cpix:ContentKey>
  <cpix:ContentKey explicitIV="L6jzdXrXAFbCJGBuMrrKrG==" kid="53abdba2-f210-43cb-bc90-
f18f9a890a02" commonEncryptionScheme="cbcs"></cpix:ContentKey>
 </cpix:ContentKeyList>
 <cpix:DRMSystemList>
  <!-- FairPlay -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="94ce86fb-07ff-4f43-adb8-93d2fa968ca2">
   <cpix:HLSSignalingData playlist="media"></cpix:HLSSignalingData>
   <cpix:HLSSignalingData playlist="master"></cpix:HLSSignalingData>
  </cpix:DRMSystem>
  <cpix:DRMSystem kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 systemId="94ce86fb-07ff-4f43-adb8-93d2fa968ca2">
   <cpix:HLSSignalingData playlist="media"></cpix:HLSSignalingData>
   <cpix:HLSSignalingData playlist="master"></cpix:HLSSignalingData>
  </cpix:DRMSystem>
  <!-- Widevine -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
```

```
    <cpix:HLSSignalingData playlist="media"></cpix:HLSSignalingData>
    <cpix:HLSSignalingData playlist="master"></cpix:HLSSignalingData>
    <cpix:ContentProtectionData></cpix:ContentProtectionData>
    <cpix:PSSH></cpix:PSSH>
   </cpix:DRMSystem>
   <cpix:DRMSystem kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
    <cpix:HLSSignalingData playlist="media"></cpix:HLSSignalingData>
    <cpix:HLSSignalingData playlist="master"></cpix:HLSSignalingData>
    <cpix:ContentProtectionData></cpix:ContentProtectionData>
    <cpix:PSSH></cpix:PSSH>
   </cpix:DRMSystem>
   <!-- Playready -->
   <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="9a04f079-9840-4286-ab92-e65be0885f95">
    <cpix:HLSSignalingData playlist="media"></cpix:HLSSignalingData>
    <cpix:HLSSignalingData playlist="master"></cpix:HLSSignalingData>
    <cpix:ContentProtectionData></cpix:ContentProtectionData>
    <cpix:PSSH></cpix:PSSH>
    <cpix:SmoothStreamingProtectionHeaderData></
cpix:SmoothStreamingProtectionHeaderData>
   </cpix:DRMSystem>
   <cpix:DRMSystem kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 systemId="9a04f079-9840-4286-ab92-e65be0885f95">
    <cpix:HLSSignalingData playlist="media"></cpix:HLSSignalingData>
    <cpix:HLSSignalingData playlist="master"></cpix:HLSSignalingData>
    <cpix:ContentProtectionData></cpix:ContentProtectionData>
    <cpix:PSSH></cpix:PSSH>
    <cpix:SmoothStreamingProtectionHeaderData></
cpix:SmoothStreamingProtectionHeaderData>
   </cpix:DRMSystem>
 </cpix:DRMSystemList>
 <cpix:ContentKeyUsageRuleList>
   <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="VIDEO">
    <cpix:VideoFilter />
   </cpix:ContentKeyUsageRule>
   <cpix:ContentKeyUsageRule kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 intendedTrackType="AUDIO">
    <cpix:AudioFilter />
   </cpix:ContentKeyUsageRule>
 </cpix:ContentKeyUsageRuleList>
</cpix:CPIX>
```

クリアでキーを含むレスポンスペイロードの VOD 例

次の例は、DRM キープロバイダーからの一般的なレスポンスペイロードを示しています (読みやすくするために、戻り値は [...] で短縮されています)。

```
<cpix:CPIX contentId="abc123" version="2.3" xmlns:cpix="urn:dashif:org:cpix"
 xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc">
 <cpix:ContentKeyList>
  <cpix:ContentKey explicitIV="OFj2IjCsPJFfMAxmQxLGPw==" kid="98ee5596-cd3e-a20d-163a-
e382420c6eff" commonEncryptionScheme="cbcs">
   <cpix:Data>
    <pskc:Secret>
     <pskc:PlainValue>5dGAgwGuUYu4dHeHtNlxJw==</pskc:PlainValue>
    </pskc:Secret>
   </cpix:Data>
  </cpix:ContentKey>
  <cpix:ContentKey explicitIV="L6jzdXrXAFbCJGBuMrrKrG==" kid="53abdba2-f210-43cb-bc90-
f18f9a890a02" commonEncryptionScheme="cbcs">
   <cpix:Data>
    <pskc:Secret>
     <pskc:PlainValue>h3toSFIlyAYpfXVQ795m6x==</pskc:PlainValue>
    </pskc:Secret>
   </cpix:Data>
  </cpix:ContentKey>
 </cpix:ContentKeyList>
 <cpix:DRMSystemList>
  <!-- FairPlay -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="94ce86fb-07ff-4f43-adb8-93d2fa968ca2">
   <cpix:HLSSignalingData playlist="media">aHR0cHM6L[...]WZm</cpix:HLSSignalingData>
   <cpix:HLSSignalingData playlist="master">Y29tLmFwc[...]XJ5</cpix:HLSSignalingData>
  </cpix:DRMSystem>
  <cpix:DRMSystem kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 systemId="94ce86fb-07ff-4f43-adb8-93d2fa968ca2">
   <cpix:HLSSignalingData playlist="media">trBAnbMcj[...]u44</cpix:HLSSignalingData>
   <cpix:HLSSignalingData playlist="master">mn626PjyR[...]2fi</cpix:HLSSignalingData>
  </cpix:DRMSystem>
  <!-- Widevine -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
   <cpix:HLSSignalingData playlist="media">Ifa2V5LWl[...]nNB</cpix:HLSSignalingData>
   <cpix:HLSSignalingData playlist="master">oIARIQeSI[...]Nd2l</cpix:HLSSignalingData>
   <cpix:ContentProtectionData>RoNd2lkZXZ[...]Nib</cpix:ContentProtectionData>
   <cpix:PSSH>AAAAanBzc[...]A==</cpix:PSSH>
```

```
  </cpix:DRMSystem>
  <cpix:DRMSystem kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
   <cpix:HLSSignalingData playlist="media">lTznjvtzL[...]GfJ</cpix:HLSSignalingData>
   <cpix:HLSSignalingData playlist="master">XgzdzQH7p[...]zeX</cpix:HLSSignalingData>
   <cpix:ContentProtectionData>TdgRnuJsZ[...]wDw</cpix:ContentProtectionData>
   <cpix:PSSH>mYZbjpWdS[...]D==</cpix:PSSH>
  </cpix:DRMSystem>
  <!-- Playready -->
  <cpix:DRMSystem kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 systemId="9a04f079-9840-4286-ab92-e65be0885f95">
   <cpix:HLSSignalingData playlist="media">HicXmbZ2m[...]4==</cpix:HLSSignalingData>
   <cpix:HLSSignalingData playlist="master">GVzdCIfa2[...]Eta</cpix:HLSSignalingData>
   <cpix:ContentProtectionData>t7WwH24FI[...]YCC</cpix:ContentProtectionData>
   <cpix:PSSH>FFFFanBzc[...]A==</cpix:PSSH>
   <cpix:SmoothStreamingProtectionHeaderData>s5RrJ12HL[...]UBB</
cpix:SmoothStreamingProtectionHeaderData>
  </cpix:DRMSystem>
  <cpix:DRMSystem kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 systemId="9a04f079-9840-4286-ab92-e65be0885f95">
   <cpix:HLSSignalingData playlist="media">BptGzwis2[...]Iej</cpix:HLSSignalingData>
   <cpix:HLSSignalingData playlist="master">3c9SXdVa0[...]MBH</cpix:HLSSignalingData>
   <cpix:ContentProtectionData>HotJCMQyc[...]GpU</cpix:ContentProtectionData>
   <cpix:PSSH>S6UD43ybN[...]f==</cpix:PSSH>
   <cpix:SmoothStreamingProtectionHeaderData>VBFUv2or0[...]JeP</
cpix:SmoothStreamingProtectionHeaderData>
  </cpix:DRMSystem>
 </cpix:DRMSystemList>
 <cpix:ContentKeyUsageRuleList>
  <cpix:ContentKeyUsageRule kid="98ee5596-cd3e-a20d-163a-e382420c6eff"
 intendedTrackType="VIDEO">
   <cpix:VideoFilter />
  </cpix:ContentKeyUsageRule>
  <cpix:ContentKeyUsageRule kid="53abdba2-f210-43cb-bc90-f18f9a890a02"
 intendedTrackType="AUDIO">
   <cpix:AudioFilter />
  </cpix:ContentKeyUsageRule>
 </cpix:ContentKeyUsageRuleList>
</cpix:CPIX>
```

SPEKE API v2 - コンテンツキーの暗号化

必要に応じて SPEKE 実装にコンテンツキーの暗号化を追加できます。コンテンツキーの暗号化は、コンテンツ自体を暗号化するだけでなく、転送用のコンテンツキーを暗号化することで、完全な end-to-end 保護を保証します。キープロバイダーにこれを実装していない場合は、トランスポートレイヤーの暗号化と強力な認証をセキュリティに使用することになります。

AWS クラウドで動作するエンクリプタにコンテンツキー暗号化を使用するには、お客様は証明書を AWS Certificate Manager にインポートし、暗号化アクティビティにその結果の証明書 ARN を使用します。エンクリプタは、証明書 ARN と ACM サービスを使用して、暗号化されたコンテンツキーを DRM キープロバイダーに提供します。

制限事項

SPEKE は、DASH-IF CPIX 仕様で指定されているコンテンツキー暗号化をサポートします。ただし、次の制限があります。

- SPEKE は、リクエストおよびレスポンスペイロードにデジタル署名検証 (XMLDSIG) をサポートしていません。
- SPEKE には 2048 ビットの RSA ベースの証明書が必要です。

これらの制限は、DASH-IF 仕様のカスタマイズと制約にも記載されています。

コンテンツキーの暗号化の実装

コンテンツの暗号化キーを提供するには、DRM キープロバイダーの実装に次の項目を含めます。

- リクエストペイロードとレスポンスペイロードで要素 `<cpix:DeliveryDataList>` を処理します。
- レスポンスペイロードの `<cpix:ContentKeyList>` に暗号化された値を入力します。

これらの要素の詳細については、DASH-IF CPIX 2.3 仕様を参照してください。

レスポンスペイロードのコンテンツキー暗号化要素の例 `<cpix:DeliveryDataList>`

```
<cpix:CPIX contentId="abc123"
    version="2.3"
    xmlns:cpix="urn:dashif:org:cpix"
    xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc">
```

```
    <cpix:DeliveryDataList>
        <cpix:DeliveryData id="<ORIGIN SERVER ID>">
            <cpix:DeliveryKey>
                <ds:X509Data>
                    <ds:X509Certificate><X.509 CERTIFICATE, BASE-64 ENCODED></
ds:X509Certificate>
                </ds:X509Data>
            </cpix:DeliveryKey>
        </cpix:DeliveryData>
    </cpix:DeliveryDataList>
    <cpix:ContentKeyList>
     ...
    </cpix:ContentKeyList>
</cpix:CPIX>
```

レスポンスペイロードのコンテンツキー暗号化要素の例 <cpix:DeliveryDataList>

```
<cpix:CPIX contentId="abc123"
    version="2.3"
    xmlns:cpix="urn:dashif:org:cpix"
    xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc">
    <cpix:DeliveryDataList>
        <cpix:DeliveryData id="<ORIGIN SERVER ID>">
            <cpix:DeliveryKey>
                <ds:X509Data>
                    <ds:X509Certificate><X.509 CERTIFICATE, BASE-64 ENCODED></
ds:X509Certificate>
                </ds:X509Data>
            </cpix:DeliveryKey>
            <cpix:DocumentKey Algorithm="http://www.w3.org/2001/04/xmlenc#aes256-cbc">
                <cpix:Data>
                    <pskc:Secret>
                        <pskc:EncryptedValue>
                            <enc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/
xmlenc#rsa-oaep-mgf1p" />
                            <enc:CipherData>
                                <enc:CipherValue><RSA CIPHER VALUE></enc:CipherValue>
                            </enc:CipherData>
                        </pskc:EncryptedValue>
                        <pskc:ValueMAC>qnei/5TsfUwDu+8bhsZrLjDRDngvmnUZD2eva7SfXWw=</
pskc:ValueMAC>
                    </pskc:Secret>
                </cpix:Data>
```

```
            </cpix:DocumentKey>
            <cpix:MACMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#hmac-
sha512">
                <cpix:Key>
                    <pskc:EncryptedValue>
                        <enc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/
xmlenc#rsa-oaep-mgf1p" />
                        <enc:CipherData>
                            <enc:CipherValue><RSA CIPHER VALUE></enc:CipherValue>
                        </enc:CipherData>
                    </pskc:EncryptedValue>
                    <pskc:ValueMAC>DGqdpHUfFKxdsO9+EWrPjtdTCVfjPLwwtzEcFC/j0xY=</
pskc:ValueMAC>
                </cpix:Key>
            </cpix:MACMethod>
        </cpix:DeliveryData>
    </cpix:DeliveryDataList>
    <cpix:ContentKeyList>
     ...
    </cpix:ContentKeyList>
</cpix:CPIX>
```

レスポンスペイロードのコンテンツキー暗号化要素の例 <cpix:ContentKeyList>

次の例は、レスポンスペイロードの <cpix:ContentKeyList> 要素で暗号化されたコンテンツキーの処理を示しています。これは <pskc:EncryptedValue> 要素を使用します。

```
<cpix:ContentKeyList>
     <cpix:ContentKey explicitIV="OFj2IjCsPJFfMAxmQxLGPw==" kid="98ee5596-cd3e-
a20d-163a-e382420c6eff" commonEncryptionScheme="cbcs">
         <cpix:Data>
             <pskc:Secret>
                 <pskc:EncryptedValue>
                     <enc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/
xmlenc#aes256-cbc" />
                     <enc:CipherData>
                         <enc:CipherValue>NJYebfvJ2TdMm3k6v
+rLNVYb0NoTJoTLBBdbpe8nmilEfp82SKa7MkqTn2lmQBPB</enc:CipherValue>
                     </enc:CipherData>
                 </pskc:EncryptedValue>
                 <pskc:ValueMAC>t9lW4WCebfS1GP+dh0IicMs+2+jnrAmfDa4WU6VGHc4=</
pskc:ValueMAC>
             </pskc:Secret>
```

```
         </cpix:Data>
     </cpix:ContentKey>
 </cpix:ContentKeyList>
```

これに対して、以下の例は、クリアキーとして暗号化されていないコンテンツキーを持つ同様のレスポンスペイロードを示しています。これは <pskc:PlainValue> 要素を使用します。

```
<cpix:ContentKeyList>
    <cpix:ContentKey explicitIV="OFj2IjCsPJFfMAxmQxLGPw==" kid="98ee5596-cd3e-
a20d-163a-e382420c6eff" commonEncryptionScheme="cbcs">
        <cpix:Data>
            <pskc:Secret>
                <pskc:PlainValue>5dGAgwGuUYu4dHeHtNlxJw==</pskc:PlainValue>
            </pskc:Secret>
        </cpix:Data>
    </cpix:ContentKey>
</cpix:ContentKeyList>
```

SPEKE API v2 - キー識別子のオーバーライド

エンクリプタは、キーを回すたびに新しいキー識別子 (KID) を作成します。リクエストで KID を DRM キープロバイダーに渡します。ほとんどの場合、キープロバイダーは同じ KID を使用して応答しますが、レスポンスの KID には異なる値を提供できます。

以下は、KID 11111111-1111-1111-1111-111111111111 のリクエスト例です。

```
<cpix:CPIX contentId="abc123" version="2.3" xmlns:cpix="urn:dashif:org:cpix"
 xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc">
 <cpix:ContentKeyList>
  <cpix:ContentKey explicitIV="OFj2IjCsPJFfMAxmQxLGPw=="
 kid="11111111-1111-1111-1111-111111111111" commonEncryptionScheme="cbcs"></
cpix:ContentKey>
 </cpix:ContentKeyList>
 <cpix:DRMSystemList>
  <!-- Widevine -->
  <cpix:DRMSystem kid="11111111-1111-1111-1111-111111111111"
 systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
   <cpix:HLSSignalingData playlist="media"></cpix:HLSSignalingData>
   <cpix:HLSSignalingData playlist="master"></cpix:HLSSignalingData>
   <cpix:ContentProtectionData></cpix:ContentProtectionData>
   <cpix:PSSH></cpix:PSSH>
  </cpix:DRMSystem>
```

```
 </cpix:DRMSystemList>
 <cpix:ContentKeyPeriodList>
  <cpix:ContentKeyPeriod id="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"
 index="1" />
 </cpix:ContentKeyPeriodList>
 <cpix:ContentKeyUsageRuleList>
  <cpix:ContentKeyUsageRule kid="11111111-1111-1111-1111-111111111111"
 intendedTrackType="VIDEO">
   <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
   <cpix:VideoFilter />
  </cpix:ContentKeyUsageRule>
 </cpix:ContentKeyUsageRuleList>
</cpix:CPIX>
```

次のレスポンスは、KID を 22222222-2222-2222-2222-222222222222 にオーバーライドします。

```
<cpix:CPIX contentId="abc123" version="2.3" xmlns:cpix="urn:dashif:org:cpix"
 xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc">
 <cpix:ContentKeyList>
  <cpix:ContentKey explicitIV="OFj2IjCsPJFfMAxmQxLGPw=="
 kid="22222222-2222-2222-2222-222222222222" commonEncryptionScheme="cbcs">
   <cpix:Data>
    <pskc:Secret>
     <pskc:PlainValue>5dGAgwGuUYu4dHeHtNlxJw==</pskc:PlainValue>
    </pskc:Secret>
   </cpix:Data>
  </cpix:ContentKey>
 </cpix:ContentKeyList>
 <cpix:DRMSystemList>
  <!-- Widevine -->
  <cpix:DRMSystem kid="22222222-2222-2222-2222-222222222222"
 systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
   <cpix:HLSSignalingData playlist="media">Ifa2V5LWl[...]nNB</cpix:HLSSignalingData>
   <cpix:HLSSignalingData playlist="master">oIARIQeSI[...]Nd2l</cpix:HLSSignalingData>
   <cpix:ContentProtectionData>RoNd2lkZXZ[...]Nib</cpix:ContentProtectionData>
   <cpix:PSSH>AAAAanBzc[...]A==</cpix:PSSH>
  </cpix:DRMSystem>
 </cpix:DRMSystemList>
 <cpix:ContentKeyPeriodList>
  <cpix:ContentKeyPeriod id="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"
 index="1" />
 </cpix:ContentKeyPeriodList>
```

```
  <cpix:ContentKeyUsageRuleList>
   <cpix:ContentKeyUsageRule kid="22222222-2222-2222-2222-222222222222"
  intendedTrackType="VIDEO">
    <cpix:KeyPeriodFilter periodId="keyPeriod_0909829f-40ff-4625-90fa-75da3e53278f"/>
    <cpix:VideoFilter />
   </cpix:ContentKeyUsageRule>
  </cpix:ContentKeyUsageRuleList>
</cpix:CPIX>
```

ライセンス

Creative Commons Attribution-ShareAlike 4.0 International Public License

ライセンスされた権利 (以下で定義) を行使することにより、お客様は、クリエイティブコモンズ表示 - ShareAlike 4.0 国際パブリックライセンス (「パブリックライセンス」) の条項に拘束されることを受諾し、同意するものとします。本パブリックライセンスが契約と解釈されるであろう範囲において、あなたはこれらの利用条件のあなたによる受諾と引き換えにライセンスされた権利を付与されます。そして、許諾者は、あなたに対し、それらの条項のもとでライセンス対象物を利用可能にすることから許諾者が受領する利益と引き換えに、そのような権利を付与します。

第 1 条 – 定義

a. 「翻案物」とは、著作権およびそれに類する権利の対象となり、ライセンス対象物について許諾者が有する著作権およびそれに類する権利に基づく許諾が必要とされるような形で、翻訳され、改変され、編集され、変形され、またはその他の方法により変更されたマテリアルで、ライセンス対象物から派生したか、またはライセンス対象物に基づくものを意味します。本パブリックライセンスにおいては、ライセンス対象物が音楽作品、実演または録音物で、これらが動画と同期させられる場合には、翻案物が常に作成されることになります。

b. 「翻案者のライセンス」とは、翻案物に対してあなたが寄与した部分に生じる、あなたの著作権およびそれに類する権利について、本パブリックライセンスの条項に従って、あなたが適用するライセンスのことをいいます。

c. BY-SA 互換ライセンスとは、クリエイティブコモンズによって本質的に本パブリックライセンスに相当するものとして承認された、creativecommons.org/compatiblelicenses に記載されているライセンスを意味します。

d. 「著作権およびそれに類する権利」とは、その権利がどのように名づけられ、または分類されるかにかかわらず、著作権および/または著作権に密接に関係する類似の権利をいいます (実演、放送、録音物、およびデータベース権を含むが、これに限られません)。本パブリックライセンスに

おいては、第 2 条 (b)(1) および (2) において規定される権利は、著作権およびそれに類する権利ではありません。

e. 「効果的な技術的保護手段」とは、1996 年 12 月 20 日に採択された WIPO 著作権条約第 11 条、および/または類似の国際協定の義務を満たす諸法規の下で、正当な権限なしに回避されてはならないものとされる諸手段をいいます。

f. 「例外および権利制限」とは、ライセンス対象物をあなたが利用する場合に適用される、フェアユース、フェアディーリングおよび/または著作権およびそれに類する権利に対するその他の例外もしくは権利制限をいいます。

g. ライセンス要素とは、クリエイティブコモンズパブリックライセンスの名前でリストされているライセンス属性を意味します。本パブリックライセンスのライセンス要素は、帰属および です ShareAlike。

h. 「ライセンス対象物」とは、許諾者が本パブリックライセンスを適用した美術的または文学的著作物、データベース、またはその他のマテリアルを意味します。

i. 「ライセンスされた権利」とは、本パブリックライセンスの条項に基づき、あなたに与えられる権利をいい、かかる権利は、あなたによるライセンス対象物の利用に適用され、かつ、許諾者がライセンスする権限を有する、全ての著作権およびそれに類する権利に限定されます。

j. 「許諾者」とは、本パブリックライセンスのもとで権利を付与する個人または団体を意味します。

k. 「共有」とは、複製、公開の展示、公開の上演・演奏、頒布、配布、通信または輸入のような、ライセンスされた権利に関する許諾を必要とするような手段または手法により、公衆に対しマテリアルを提供すること、および、公衆がマテリアルを利用できるようにすること (公衆の各人が、自ら独自に場所および時間を選択してマテリアルにアクセスすることができる方法を含みます) を意味します。

l. 「データベース権」とは、データベースの法的保護に関する 1996 年 3 月 11 日の欧州議会および理事会指令 96/9/EC の結果として生じた、著作権以外の権利、(この指令が修正および/または継承された場合それらを反映したもの)、および、世界中の本質的に同等な権利を意味します。

m. 「あなた」とは、本パブリックライセンスのもとでライセンスされた権利を行使する個人または団体をいいます。「あなたの」もそれに対応した意味となります。

第 2 条 – 範囲

a. ライセンス許諾。

1. 本パブリックライセンスの条項に従い、許諾者はあなたに対し、ライセンス対象物について、以下に掲げるライセンスされた権利を行使できる全世界的な、無償、再許諾不可、非排他的、かつ取消不可なライセンスを付与します:

   A. ライセンス対象物の全部または一部を、複製および共有すること、ならびに

   B. 翻案物を作成、複製および共有すること

2. 例外および権利制限 誤解を避けるために記すと、例外および権利制限があなたの利用に適用される部分については、本パブリックライセンスは適用されず、あなたは本パブリックライセンスの条項に従う必要はありません。

3. 期間。本パブリックライセンスの有効期間は第 6 条 (a) にて規定されます。

4. 媒体および形式、許可される技術的改変 許諾者は、あなたに対し、あらゆる媒体や形式 (現在知られているか、または今後作られるか否かを問いません) において、ライセンスされた権利を行使する権限、およびその行使に必要とされる技術的な改変を行う権限を付与します。許諾者は、あなたが、ライセンスされた権利を行使するために必要とされる技術的な改変 (効果的な技術的保護手段を回避するために必要とされる技術的な改変を含みます) を禁止するいかなる権利または権限を放棄し、および/またはこれらの権利または権限を行使しないことに同意します。本パブリックライセンスにおいては、本第 2 条 (a)(4) により認められる改変をするだけでは翻案物を作り出すことにはなりません。

5. ダウンストリーム (下流側) の受取人

   A. 許諾者からの申し出 – ライセンス対象物 ライセンス対象物の受取人は、許諾者から本パブリックライセンスの条項の下でライセンスされた権利を行使できるという申出を自動的に受け取ります。

   B. 許諾者からの追加の申し出 – 翻案物 翻案物の受領者は、許諾者から、あなたが申請した翻案者のライセンスの条件下で翻案物に対するライセンスされた権利を行使できるという申出を自動的に受け取ります。

   C. ダウンストリーム (下流側) への制限の禁止 あなたは、ライセンス対象物の受取人がライセンスされた権利を行使するのを制限されることになる場合には、ライセンス対象物に対して、いかなる追加条項または異なる条項も提案または課してはならず、あるいは、いかなる効果的な技術的保護手段も適用してはなりません。

6. 支持表明がないこと 許諾者または第 3 条 3(a)(1)(A)(i) に定められている許諾者以外のクレジット表示の対象として指定されている者が、あなたまたはライセンス対象物のあなたによる利用について、関連している、援助· 支持している、あるいは正式な地位を付与している、と主張または示唆することを本パブリックライセンスは許諾しておらず、またはそのように解釈されてはなりません。

b. その他の権利。

1. 同一性保持の権利のような著作者人格権は、本パブリックライセンスのもとではライセンスされません。パブリシティ権、プライバシー権、および/または他の類似した人格権も同様です。ただし、可能なかぎり、許諾者は、あなたがライセンスされた権利を行使するために必要とされる範囲内で、また、その範囲内でのみ、許諾者の保持する、いかなるそのような権利を放棄し、および/または主張しないことに同意します。
2. 特許権および商標権は本パブリックライセンスのもとではライセンスされません。
3. 可能なかぎり、許諾者は、ライセンスされた権利の行使について、直接か、または任意のもしくは放棄可能な法定のもしくは強制的なライセンスに関する仕組みに基づく集中管理団体を介するかを問わず、あなたからライセンス料を得るいかなる権利も放棄します。その他一切の場合において、許諾者はそのようなライセンス料を得るいかなる権利も明確に保持します。

第 3 条 – ライセンス利用条件

ライセンスされた権利をあなたが行使するにあたっては、以下に記載された諸条件に従う必要があります。

a. 表示。

1. あなたがライセンス対象物 (変更されたものを含む) を共有する場合は以下のことを行う必要があります:

   A. ライセンス対象物と共に許諾者から提供されていれば、以下のものを保持すること。

```
i . identification of the creator(s) of the Licensed Material and any others
 designated to receive attribution, in any reasonable manner requested by the
 Licensor (including by pseudonym if designated);
```

```
ii . a copyright notice;
```

```
iii . a notice that refers to this Public License;
```

```
iv . a notice that refers to the disclaimer of warranties;
```

```
v . a URI or hyperlink to the Licensed Material to the extent reasonably
 practicable;
```

   B. ライセンス対象物を改変した場合はその旨を記し、従前の改変点についての表示も保持すること。

C. ライセンス対象物が本パブリックライセンスに基づきライセンスされていることを示すこと、および、本パブリックライセンスの全文またはその URI か本パブリックライセンスへのハイパーリンクのいずれかを含めること。

2. 第 3 条 (a)(1) の条件は、あなたがライセンス対象物を共有する媒体・方法・文脈に照らして、いかなる合理的な方法でも満たすことができます。例えば、必要とされる情報を含むリソースの URI やハイパーリンクを付すことで条件を満たすことが合理的な場合があります。

3. 許諾者からリクエストされれば、あなたは第 3 条 (a)(1)(A) に掲げるいかなる情報も合理的に実施可能な範囲で削除しなければなりません。

b. ShareAlike。第 3 条 (a) の条件に加えて、あなたが作成した翻案物を共有する場合、以下の条件も適用されます。

1. あなたが申請する翻案者のライセンスは、同じライセンス要素を持つクリエイティブコモンズライセンスの本バージョン以降、または BY-SA 互換ライセンスである必要があります。

2. 申請する翻案者のライセンスのテキスト、その URI またはハイパーリンクを含める必要があります。この条件は、あなたが翻案物を共有する媒体・方法・文脈に照らして、いかなる合理的な方法でも満たすことができます。

3. あなたは、あなたが申請する翻案者のライセンスに基づいて付与された権利を行使することを制限するいかなる追加のまたは異なる条項も、翻案物に対して提案または課してはならず、あるいは、いかなる効果的な技術的保護手段も適用してはなりません。

第 4 条 – データベース権

ライセンスされた権利にデータベース権が含まれており、ライセンス対象物のあなたの利用に適用される場合:

a. 誤解を避けるために記すと、第 2 条 (a)(1) に従い、データベースの全てまたは実質的な部分のコンテンツの抽出、再利用、複製または共有をする権利をあなたに与えます。

b. あなたがデータベース権を持つデータベースに、あなたが、本データベースのコンテンツの全てまたは実質的な部分を含める場合、あなたがデータベース権を持つデータベース (ただし、個々のコンテンツではありません) は、本第 3 条 (b) の目的を含めて、翻案物となります。

c. あなたは、データベースのコンテンツの全てまたは実質的な部分を共有する場合は、第 3 条 (a) の条件に従わなくてはなりません。誤解を避けるために記すと、本第 4 条は、ライセンスされた権利が他の著作権およびそれに類する権利を含む場合の本パブリックライセンス下でのあなたの義務に追加されるものであり、置き換えるものではありません。

第 5 条 – 無保証および責任制限

a. 許諾者が別途合意しない限り、許諾者は可能な範囲において、ライセンス対象物を現状有姿のまま、現在可能な限りで提供し、明示、黙示、法令上、その他に関わらずライセンス対象物について一切の表明または保証をしません。これには、権利の帰属、商品性、特定の利用目的への適合性、権利侵害の不存在、隠れた瑕疵その他の瑕疵の不存在、正確性または誤りの存在もしくは不存在を含みますが、これに限られず、既知であるか否か、発見可能であるか否かを問いません。全部または一部の無保証が認められない場合、この無保証はあなたには適用されないこともあります。

b. 可能な範囲において、本パブリック・ライセンスもしくはライセンス対象物の利用によって起きうる直接、特別、間接、偶発、結果的、懲罰的その他の損失、コスト、出費または損害について、例え損失、コスト、出費、損害の可能性について許諾者が知らされていたとしても、許諾者は、あなたに対し、いかなる法理 (過失を含みますがこれに限られません) その他に基づいても責任を負いません。全部または一部の責任制限が認められない場合、この制限はあなたには適用されないこともあります。

c. 上記の無保証および責任制限は、可能な範囲において、全責任の完全な免責および免除に最も近いものとして解釈するものとします。

第 6 条 – 期間および終了

a. 本パブリックライセンスは、ここでライセンスされた著作権およびそれに類する権利が有効な期間、適用されます。ただし、あなたが本パブリックライセンスに違反した場合、本パブリックライセンスに定めるあなたの権利は自動的に終了します。

b. ライセンス対象物をあなたが利用する権利が第 6 条 (a) の事由により終了した場合でも:

   1. あなたが違反を発見してから 30 日以内に違反を是正した場合に限り、違反を是正したその日に、自動的に復活します。または、

   2. 許諾者により権利の復活を明示された場合に、復活します。

c. 誤解を避けるために記すと、本第 6 条 (b) は、許諾者が、あなたの本パブリックライセンスに関する違反に対する救済を求めるために有するであろういかなる権利にも影響を及ぼしません。

d. 誤解を避けるために記すと、許諾者は、いつでも、別の条項の下でライセンス対象物を提供したり、ライセンス対象物の配布を停止することができます。しかし、その場合でも、本パブリックライセンスは終了しません。

e. 第 1 条、第 5 条、第 6 条、第 7 条、第 8 条は、本パブリックライセンスが終了してもなお有効に存続します。

第 7 条 – その他の条項

a. 許諾者は、明確に合意しない限り、あなたが通知するいかなる追加のまたは異なる条項にも拘束されません。

b. ライセンス対象物に関する取り決め、了解事項または合意でここに言明されていない一切のものは、本パブリックライセンスの条項とは切り離され、独立したものです。

第 8 条 – 解釈

a. 誤解を避けるために記すと、本パブリックライセンスは、本パブリックライセンスによる許諾に基づかない、ライセンス対象物のいかなる合法的な利用も縮小したり、限定したり、制限したり、条件を課したりするものではなく、またそのように解釈されてはなりません。

b. 可能な範囲で、本パブリックライセンスのいずれかの規定が執行不能とみなされた場合には、本パブリックライセンスは、執行可能とするために必要最小限度の範囲で自動的に変更されます。もしある規定の変更が不可能な場合には、その他の条項の執行可能性に影響を与えることなく、当該規定は本パブリックライセンスから切り離されます。

c. 本パブリックライセンスのいかなる条項も、許諾者の明確な合意なしには、放棄されることはなく、また、順守しないことに同意することはありません。

d. 本パブリックライセンスのいかなる条項も、許諾者やあなたに適用される、あらゆる特権や免責 (司法権や当局の法的手続からの特権や免責を含む) に対する制限や放棄を構成するものではなく、またそのように解釈されるものではありません。

ドキュメント履歴

以下の表は SPEKE のドキュメントの変更点をまとめたものです。

SPEKE v1

変更	説明	日付
サポートマトリックス: AWS パートナーのサービスと製品	AWS パートナーのサービスと製品で、SPEKE サポートに関する新しいセクションを追加し、Bitmovin サービスを一覧表示しました。	2023 年 1 月 13 日
DRM プラットフォームプロバイダーの更新	DRM プラットフォームプロバイダーリストにリンクと新しいパートナー情報を追加しました。	2019 年 1 月 24 日
サードパーティーのエンクリプタを含める	サードパーティーエンクリプタに対応するようにアーキテクチャと説明を更新しました。	2018 年 11 月 20 日
コンテンツキー暗号化	コンテンツキーを暗号化するオプションが追加されました。これ以前は、Secure Packager and Encoder Key はクリアキーの配信のみをサポートしていました。	2018 年 10 月 30 日
サポートマトリックス - AWS Elemental Live	AWS Elemental Live マトリックスサポートを追加しました。	2018 年 9 月 27 日
スタンダードペイロードコンポーネント	JSON ペイロードの主な要素を定義するセクションを追加しました。	2018 年 9 月 27 日

変更	説明	日付
KID のオーバーライド	キープロバイダーによる KID オーバーライドに関するセクションを追加しました。	2018 年 9 月 27 日
DASH-IF サイトへのリンクを修正しました	CPIX 仕様およびシステム ID ページの DASH IF サイトのリンクを修正しました。	2018 年 9 月 27 日
AWS Elemental Live のコピーをリリース	AWS Elemental 製品 が含まれるように SPEKE ドキュメントを更新しました。	2018 年 7 月 20 日
CMAF	一般的メディアアプリケーションフォーマット (CMAF、Common Media Application Format) を含めるようにサービスのサポートマトリックステーブルを更新しました。	2018 年 6 月 27 日
初回リリース	Secure Packager and Encoder Key Exchange (SPEKE) バージョン 1 の最初のリリース。コンテンツの暗号化サービスと DRM キープロバイダー間の通信用の仕様。受信キーリクエストを処理するために、DRM キープロバイダーは Secure Packager and Encoder Key Exchange API を公開します。	2017 年 11 月 27 日

SPEKE v2

変更	説明	日付
DRM プラットフォームプロバイダーの更新のセクション	DRM プラットフォームプロバイダーリストの SPEKE v2 の列に新しい認定パートナーを追加しました。	2023 年 8 月 9 日
Live および VOD ワークフローメソッド呼び出しの例の更新のセクション	SPEKE v2 Live および VOD ワークフローメソッド呼び出しの例のセクションに欠落していた X-Speke-Version レスポンスヘッダーを追加しました。	2023 年 1 月 13 日
DRM プラットフォームプロバイダーと暗号化契約の更新のセクション	DRM プラットフォームプロバイダーリストの SPEKE v2 の列に新しい認定パートナーを追加しました。暗号化契約の新しい例を 2 つ追加し、関連するすべての例で SD の最大解像度を 1024x576 に変更しました。	2022 年 1 月 27 日
初回リリース	Secure Packager and Encoder Key Exchange (SPEKE) バージョン 2.0 の最初のリリース。コンテンツの暗号化サービスと DRM キープロバイダー間の通信用の仕様。受信キーリクエストを処理するために、DRM キープロバイダーは Secure Packager and Encoder Key Exchange API を公開します。	2021 年 9 月 7 日

AWS 用語集

AWS の最新の用語については、「AWS の用語集リファレンス」の「AWS 用語集」を参照してください。

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。