PallyCon 라이선스 토큰 가이드 v1.0

개요 {#intro}

PallyCon 클라우드 서버에서 멀티 DRM(FPS, Widevine, PlayReady, NCG) 라이선스를 발급하는 방식은 콜백 방식과 토큰 방식이 있습니다.

1. 콜백 방식

- PallyCon 클라우드 서버는 멀티 DRM 클라이언트로부터 라이선스 요청을 받으면 먼저 해당 사용자가 유효한 권한이 있는지 해당 서비스 사이트의 콜백 페이지를 통해 확인합니다.
- 권한이 있는 사용자로부터의 요청인 경우, 서비스 사이트는 콜백 웹 페이지를 통해 인증 여부, 사용 권한(무제한, 기간제), 각종 보안 옵션 등의 정보를 PallyCon 클라우드 서버에게 리턴합니다.
- PallyCon 클라우드 서버는 콜백 페이지의 응답을 받아 클라이언트에 해당 라이선스를 발급합니다.

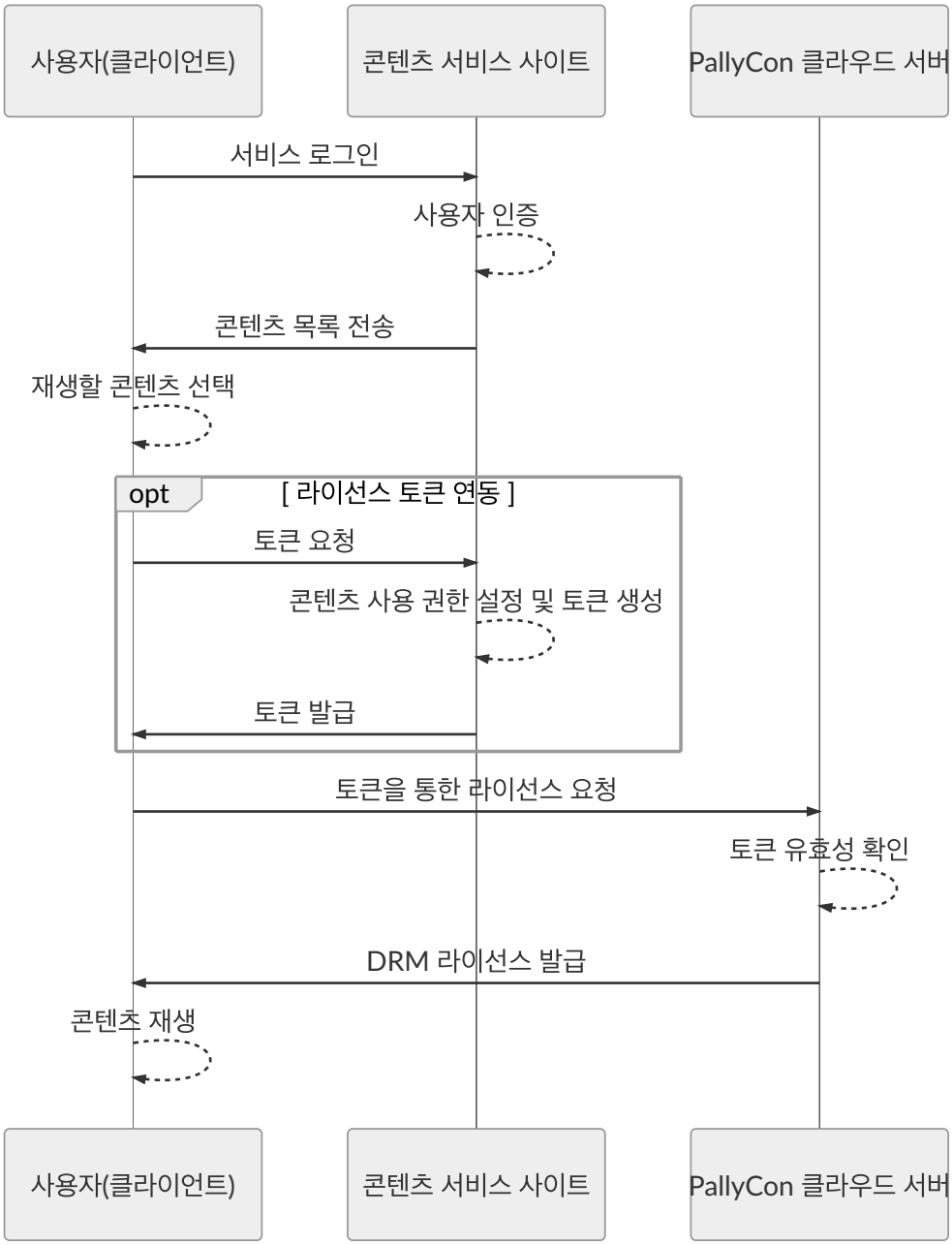
2. 토큰 방식

- 멀티 DRM 클라이언트에서 콘텐츠 재생 시, 라이선스 발급을 위해 우선 서비스 사이트에 토큰을 요청합니다. 서비스 사이트는 토큰을 요청한 사용자가 콘텐츠에 대한 권한이 있는지 확인한 후, 생성 규칙에 따라 토큰 데이터를 생성해 클라이언트에 전달합니다.
- 서비스 사이트는 토큰 데이터 내부에 콘텐츠의 사용 권한(무제한, 기간제)과 각종 보안 옵션 등을 설정할 수 있습니다.
- 클라이언트가 토큰과 함께 라이선스를 요청하면, PallyCon 클라우드 서버는 해당 토큰의 유효성을 확인한 후 라이선스를 발급합니다.

본 문서는 두 번째 방식인 토큰 방식 라이선스 발급에 사용되는 토큰의 생성 규격에 대해 설명합니다. 콜백 방식 규격에 대해서는 라이선스 콜백 가이드를 참고하시기 바랍니다.

> 토큰 생성 방법에 대한 단계 별 상세 안내와 샘플 코드는 라이선스 토큰 튜토리얼 문서에서 확인할 수 있습니다.

토큰 방식 라이선스 발급 과정 {#workflow}



(1) 토큰 요청

- 클라이언트는 재생하려는 콘텐츠의 DRM 라이선스 발급에 필요한 토큰을 서비스 사이트에 요청합니다.

(2) 토큰 생성 (아래 규격 참조)

- 서비스 사이트는 클라이언트에서 받은 요청을 확인하고, 해당 사용자가 콘텐츠 사용 권한이 있는 경우 토큰을 생성합니다.
- 토큰에는 Content ID, 사용자 ID, 타임 스템프, 라이선스 룰 정보 등이 포함됩니다.

(3) 토큰 전달

- 서비스 사이트는 위 과정에서 생성된 토큰을 요청한 클라이언트에 전달합니다.

> 토큰의 생성 및 요청/전달 기능은 PallyCon SDK 제품으로 제공되지 않으며, 서비스 사이트 측에서 본 규격을 참고해 직접 구현하셔야 합니다.

(4) 라이선스 요청

- 클라이언트는 서비스 사이트로부터 전달받은 토큰(base64 문자열)을 pallycon-customdata-v2에 담아 PallyCon 클라우드 서버에 라이선스를 요청합니다.

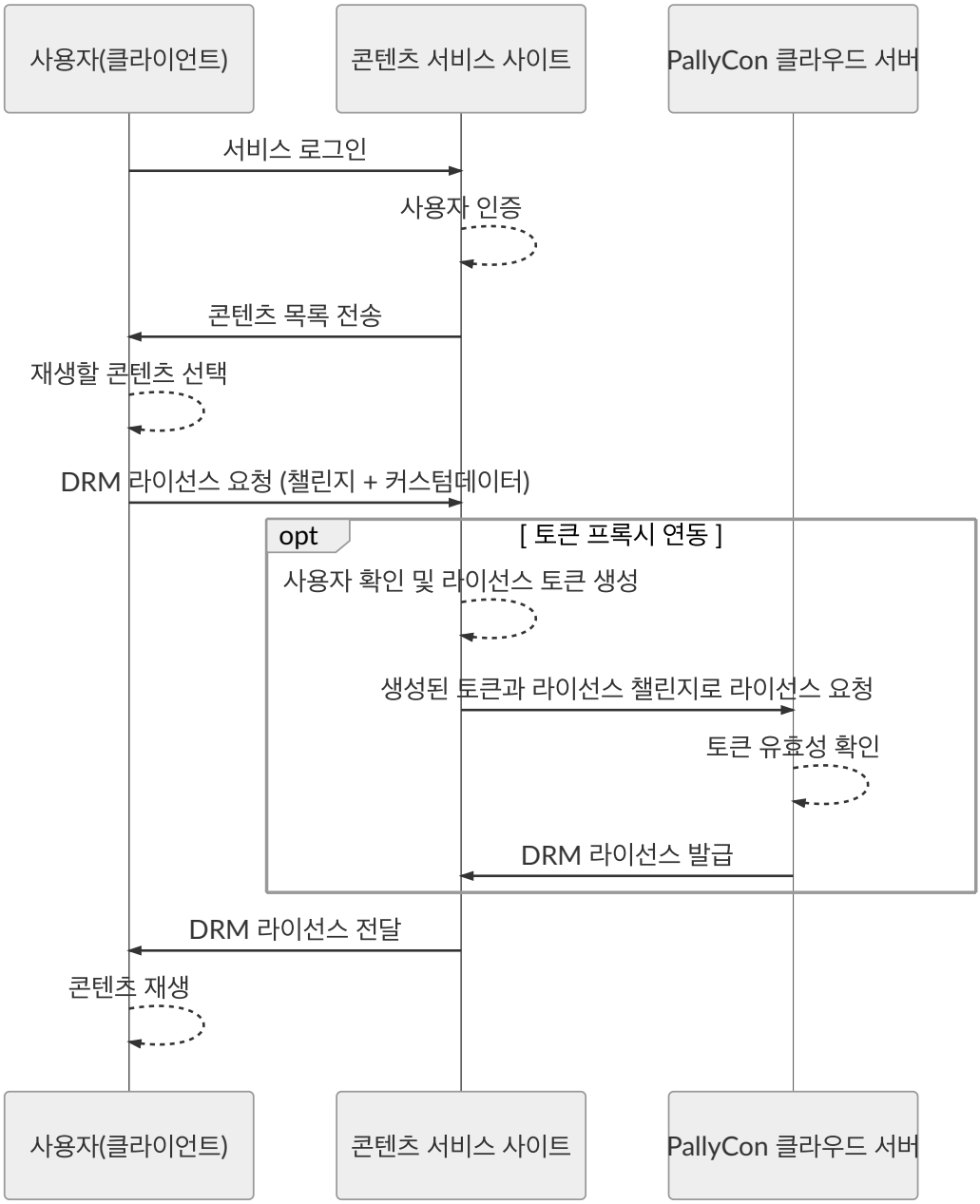
> CustomData를 통한 라이선스 요청은 멀티 DRM 라이선스 연동 가이드 또는 클라이언트 연동 가이드를 참조하시기 바랍니다.

(5) 라이선스 발급

- PallyCon 클라우드 서버는 해당 토큰의 유효성을 확인하고 설정된 룰에 따라 라이선스를 발급합니다.

토큰 프록시 방식 라이선스 발급 과정

토큰 방식 라이선스 발급은 아래와 같이 서비스 사이트의 프록시 서버를 통해 처리할 수도 있습니다.



(1) 프록시 서버로 라이선스 요청

- 클라이언트(Player)에서 DRM 콘텐츠 재생을 위해 서비스 사이트의 프록시 서버로 DRM 라이선스를 요청합니다.
- DRM LA_URL 설정에 PallyCon 라이선스 서버 URL 대신 프록시 서버의 URL을 입력하며, 커스텀 헤더 또는 URL 파라미터 등의 방식으로 라이선스 발급에 필요한 User ID와 Content ID를 전달합니다.
- 프록시 서버로 전달되는 라이선스 요청에는 클라이언트의 DRM 모듈에서 생성된 챌린지 데이터가 포함됩니다.

(2) 사용자 권한 확인 및 토큰 생성

- 서비스 사이트는 프록시 서버로 전달된 사용자와 콘텐츠 정보를 이용해 해당 콘텐츠에 권한이 있는 사용자인지 확인합니다.

- 프록시 서버는 서비스 사이트의 비즈니스 모델 및 콘텐츠 보안 정책에 따라 DRM 라이선스 규칙을 설정하여 토큰을 생성합니다.

(3) 라이선스 요청 및 발급

- 프록시 서버는 생성된 토큰을 클라이언트에서 보내온 라이선스 챌린지 데이터와 함께 PallyCon 라이선스 서버에 전송해 DRM 라이선스를 요청합니다.
- PallyCon 서버는 토큰의 유효성을 확인하고 챌린지 데이터를 이용해 DRM 라이선스를 발급합니다.

(4) 라이선스 전달 및 재생

- 프록시 서버는 PallyCon 라이선스 서버로부터 발급받은 DRM 라이선스 데이터를 클라이언트에 전달합니다.
- 클라이언트 플레이어는 전달된 DRM 라이선스를 이용해 콘텐츠 재생을 시작합니다.

토큰 생성 규격 {#token-json}

- 서비스 사이트는 클라이언트로부터의 요청에 대하여 아래와 같은 JSON 값을 생성하고, 해당 값을 Base64로 인코딩한 문자열을 응답으로 전달합니다.
- PallyCon 사이트의 DevConsole 페이지에 아래 규격의 주요 값들을 입력하여 쉽게 토큰을 생성해볼 수 있습니다.

토큰 JSON 형식

```
{
    "drm_type": "<DRM 종류>",
    "site_id": "<사이트 ID>",
    "user_id": "<사용자 ID>",
    "cid": "<콘텐트 ID>",
    "policy": "<base64(aes256(라이선스 발급 관련 토큰 룰 데이터))>",
    "timestamp": "<yyyy-mm-ddThh:mm:ssZ 형식의 토큰 유효성 발효 시간(GMT)>",
    "hash": "<base64(sha256(해시 메시지))>"
}
```

Name	Value	Required	Description
drm_type	string	No	DRM 종류 (“NCG”, “Widevine”, “PlayReady”, “FairPlay”), 기본값 : “PlayReady”

Name	Value	Required	Description
site_id	string	Yes	PallyCon 서비스에서 발급받은 서비스 사이트 ID
user_id	string	Yes	라이선스를 요청하는 사용자의 ID (서비스 사이트에서 관리하는 유저 ID). 입력할 사용자 ID가 없을 경우, 기본값 (“LICENSETOKEN”) 입력
cid	string	Yes	재생하려는 콘텐츠의 고유 ID. DRM 패키징 시에 사용된 값 (최대 200바이트 영숫자)
policy	base64 encoded string	Yes	라이선스 발급 관련 토큰 룰 데이터(상세 규격 참조)를 AES256 암호화하고 Base64로 인코딩한 문자열
timestamp	string	Yes	토큰 유효성 발효 시간(GMT 기준) ‘yyyy-mm-ddThh:mm:ssZ’. 일반적으로 현재 시간을 설정하며, 이후 기본 600초 동안 토큰이 유효함. (콘솔 사이트에서 설정 가능)
hash	base64 encoded string	Yes	해시 메시지(상세 규격 참조)를 SHA256으로 해시하고 Base64로 인코딩한 문자열

토큰 JSON 예제


```
{
    "drm_type":"Widevine",
    "site_id":"ABCD",
    "user_id":"LICENSETOKEN",
    "cid":"sample-content-id-0123",
    "policy":"uZ0ALHJDHdZKc9pICii6Hog46frSIl+to/Wbf08uqliQVjGwK0Lw40onRM743
    "timestamp":"2018-04-14T23:59:59Z",
    "hash":"QkM4NDVGMDMxRUE4MDM0NUMzQUE4MTgyMTA4QTQ2QjQyNEFBNTJCNkQ1QjhGODg
}
```


> 참고
>
> 위 예제에 사용된 값들은 실제 동작하는 데이터가 아닌 참고용이며, 실제 적용 시에는 본 규격에 따라 생성된 값을 사용해야 합니다.

최종 토큰 문자열 예제

- 위 예제의 토큰 JSON을 Base64 인코딩한 최종 토큰 문자열은 아래와 같습니다.

```
ewogICAg4oCcZHJtX3R5cGXigJ064oCdV2lkZXZpbmXigJ0sCiAgICDigJxzaXRlX2lk4oCdOuK
```

토큰 룰 JSON 형식 {#token-policy-json}

- 아래 형식으로 구성된 JSON 값을 AES256으로 암호화하고, 해당 결과를 Base64로 인코딩한 문자열을 토큰 JSON의 'token' 값으로 사용합니다.
- AES256 암호화 방식은 해당 규격을 참고하시기 바랍니다.

```
{
    "playback_policy": {
        "limit": <true|false>,
        "persistent": <true|false>,
        "duration" : <int(seconds)>,
        "expire_date": "<yyyy-mm-ddThh:mm:ssZ 형식의 만료 시간(GMT)>"
    },
    "security_policy": {
        "hardware_drm": <true|false>,
        "output_protect": {
            "allow_external_display" : <true|false>,
            "control_hdcp": <0|1|2>
        },
        "allow_mobile_abnormal_device" : <true|false>,
        "playready_security_level": <150|2000>
    },
    "external_key": {
        "mpeg_cenc": {
            "key_id" : "<hex-string>",
            "key" : "<hex-string>",
            "iv" : "<hex-string>"
        },
        "hls_aes" : {
            "key" : "<hex-string>",
            "iv" : "<hex-string>"
        },
        "ncg":{
            "cek":"<hex-string>"
        }
    }
}
```

Name	Value	Required	Description
playback_policy	json	No	재생과 관련된 룰 설정 (상세 규격 참조)
security_policy	json	No	보안 관련 룰 설정 (상세 규격 참조)
external_key	json	No	재생할 콘텐츠의 키 정보 (상세 규격 참조) 키 정보를 PallyCon 클라우드 서버에 저장하지 않고 별도로 관리하는 경우에 사용합니다.

playback_policy {#playback-policy}

Name	Value	Required	Description
limit	boolean	No	기간제 적용 여부 (기본값: false) true : 기간제, false : 무제한
persistent	boolean	No	오프라인용 라이선스 저장 여부. (기본값: false) true : 라이선스 유지, false : 재생 후 라이선스 제거(스트리밍)
duration	number	No	라이선스 유효 기간 (단위 : 초). **duration, expire_date 동시 설정 시, duration 값을 우선시 합니다.** 'limit'값이 true인 경우에만 적용됩니다. (false일 경우 이 항목은 무시됨)
expire_date	string	No	라이선스 만료 날짜, GMT Time 표기'yyyy-mm-ddThh:mm:ssZ' 'limit'값이 true인 경우에만 적용됩니다. (false일 경우 이 항목은 무시) duration 항목과 함께 사용된 경우, 이 항목은 무시됩니다.

security_policy {#security-policy}

Name	Value	Required	Description

Name	Value	Required	Description
hardware_drm	boolean	No	하드웨어 DRM 적용 여부. (기본값 false) - 각 DRM 종류에 따라 security level 설정됨 - cenc (Widevine Modular) 컨텐츠에만 적용
output_protect	json	No	외부 출력 룰 설정 (상세 규격 참조)
allow_mobile_abnormal_device	boolean	No	탈옥 기기 재생 허용 여부 (기본값 false)
playready_security_level	number	No	PlayReady 보안 레벨 설정, 150,2000 (기본값 150)

security_policy.output_protect {#output-protect}

Name	Value	Required	Description
allow_external_display	boolean	No	모바일 외부 출력 허용 여부. (기본값 false) NCG DRM에만 해당
control_hdcp	number	No	HDCP 적용 여부. (기본값 0) 0 : HDCP 제어 안함, 1 : HDCP 1.4 필요, 2 : HDCP 2.2 필요

external_key {#external-key}

Name	Value	Required	Description
mpeg_cenc	json	No	CENC 외부 키 정보 설정 - PlayReady/Widevine (상세 규격 참조)
hls_aes	json	No	HLS AES 외부 키 정보 설정 - FairPlay Streaming (상세 규격 참조)
ncg	json	No	NCG 외부 키 정보 설정 (상세 규격 참조)

external_key.mpeg_cenc {#external-key-cenc}

Name	Value	Required	Description
key_id	hex-string	No	DASH CENC 패키징(PlayReady/Widevine) 시 사용한 key ID 16byte hex string 값
key	hex-string	No	DASH CENC 패키징 시 사용한 key 16byte hex string 값
iv	hex-string	No	DASH CENC 패키징 시 사용한 iv 16byte hex string 값

external_key.hls_aes {#external-key-aes}

Name	Value	Required	Description
key	hex-string	No	HLS Sample AES 패키징(FairPlay Streaming) 시 사용한 key 16byte hex string 값
iv	hex-string	No	HLS Sample AES 패키징 시 사용한 iv 16byte hex string 값

external_key.ncg {#external-key-ncg}

Name	Value	Required	Description
cek	hex-string	No	NCG 패키징 시 사용한 cek 32byte hex string 값

토큰 룰 JSON 예제

기본적인 연동 테스트를 위한 토큰 룰은 아래 예제를 참고하시기 바랍니다. 아래 룰은 최대 재생 시간 5분인 스트리밍용 라이선스를 생성합니다.

```
{
    "playback_policy": {
        "limit": true,
        "persistent": false,
        "duration" : 300
    }
}
```

아래는 보다 상세한 설정이 추가된 풀 스펙의 예제입니다.

```
{
    "playback_policy":{
        "limit":true,
        "persistent":true,
        "duration":3600,
        "expire_date":"2018-04-20T23:59:59Z"
    },
    "security_policy":{
        "hardware_drm":true,
        "output_protect": {
            "allow_external_display":false,
            "control_hdcp":1
        },
        "allow_mobile_abnormal_device":false,
        "playready_security_level":150
    },
    "external_key": {
        "mpeg_cenc": {
            "key_id" : "30313233343536373839616263646566",
            "key" : "30313233343536373839616263646566",
            "iv" : "30313233343536373839616263646566"
        },
        "hls_aes" : {
            "key" : "30313233343536373839616263646566",
            "iv" : "30313233343536373839616263646566"
        },
        "ncg":{
            "cek":"303132333435363738396162636465663031323334353637383961626
        }
    }
}
```

SHA256 해시 메시지 형식 {#hash-message}

- 해시 메시지는 전체 토큰 JSON 데이터의 무결성을 검증하기 위해 사용되며, 아래와 같은 방법으로 생성되어야 합니다.

```
base64( sha256( <site access key> + <drm type> + <site id> + <user id> + <c
```

1. 서비스 사이트의 access key와 토큰 JSON 데이터의 'hash' 필드를 제외한 값들을 순서대로 연결시킨 문자열을 생성합니다. access key는 PallyCon 콘솔 사이트에 로그인하여 확인 할 수 있습니다.
2. 위에서 생성된 문자열의 sha256 해시값을 base64 인코딩하여 최종 해시 메시지 문자열을 생성합니다.

> sha256 해시 함수의 결과 값은 문자열로 변환하지 않고 바이트 배열 형태 그대로 base64 함수에 입력되어야 합니다.

SHA256 해시 메시지 예제

```
1단계 원본 문자열
<Access Key>WidevineABCDLICENSETOKENsample-centent-id-0123uZ0ALHJDHdZKc9pIC

2단계 sha256 + base64 문자열 (최종 형태)
QkM4NDVGMDMxRUE4MDM0NUMzQUE4MTgyMTA4QTQ2QjQyNEFBNTJCNkQ1QjhGODg1NUE1MDI2NjQ
```

AES256 암호화 {#aes256}

- AES256 암호화/복호화 처리는 PallyCon 콘솔 사이트에서 서비스 사이트 생성 시 발급 되는 사이트 키 값을 이용하여 아래와 같이 처리합니다. (PallyCon 콘솔 사이트의 셋팅 페이지에서 확인)
- PallyCon 사이트의 DevConsole 페이지에서 AES256 암호화/복호화를 테스트할 수 있습니다.

```
- mode : CBC
- key : 32 byte (PallyCon 서비스 사이트에서 발급된 사이트 키)
- iv : 16 byte ( 0123456789abcdef )
- padding : pkcs7
```
