

## III. 스토리지/CDN

### 4. CDN Standard

#### 목차

- 4.1 CDN Standard 서비스 소개
- 4.2 CDN Standard FAQ
- 4.3 CDN Standard 이용방법
- 4.4 CDN Standard 기능

## 5.1 ucloud CDN Standard 서비스 소개

### 5.1.1 목적/용도

#### □ CDN(Content Network Delivery) 이란?

인터넷 사용자들에게 웹 콘텐츠를 빠르고 안정적으로 전달하기 위해서 등장한 서비스로서 다수의 CDN 서버(Edge Server)를 구축하여 사용자가 가까운 위치에 있는 CDN 서버에서 콘텐츠를 다운로드 할 수 있도록 함으로써 콘텐츠의 전송 속도를 향상시키고 다수의 CDN 서버가 지속적으로 유지 되어 안정적인 서비스를 할 수 있는 서비스입니다.

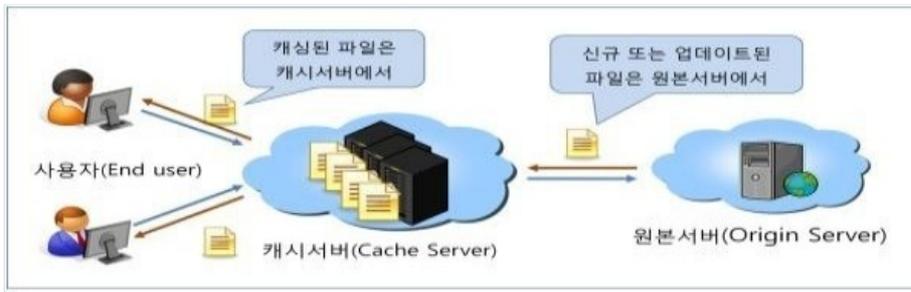
ucloud CDN 서비스는 CDN 서비스에 적합한 웹 캐시서버(Web Cache Server)를 구성하여 빠른 응답 속도와 반복적인 요청에 의한 부하 감소와 트래픽 절감 효과를 볼 수 있도록 하였습니다.

#### □ 서비스 특징점

- End-User의 체감 품질(응답속도, 가용성, 확장성, 보안 등) 향상

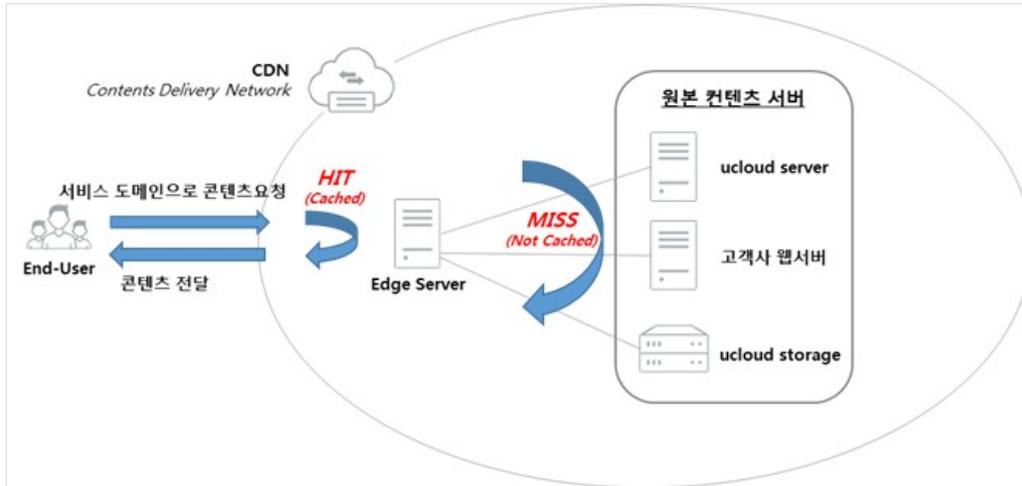
#### □ 웹 캐시 서비스(Web Cache Service)란?

웹 캐시 서비스는 이전 사용자(End User)의 요청에 의해 사용된 웹 콘텐츠를 저장하고 있다가, 재요청이 있을 경우 원본서버(Origin Server)에 요청하지 않고 이전에 저장한 콘텐츠를 직접 전송함으로써 지연 시간을 최소화하여 신속한 웹 서비스를 제공하며, 반복되는 데이터 전송을 최소화 함으로써 원본서버의 부하를 감소시키고 네트워크 대역폭을 절약할 수 있는 서비스 입니다.



## 5.1.2 구조/원리

### □ 시스템 구성도



## 5.1.3 유의사항/제약사항

### □ 타사 CDN 서비스 -> ucloud CDN Standard 서비스를 이용하는 경우

기존에 타사 CDN 서비스를 이용하시다가 ucloud CDN 서비스를 이용하실 때는 원본서버의 요청수와 트래픽이 발생할 수 있음을 유념을 하셔야 합니다.

기존에 타사에서 이용하시던 CDN 서비스에서는 많은 콘텐츠가 캐시된 상태에서 서비스를 하고 있었으므로 원본서버로의 요청을 많이 하지 않았지만 새로운 CDN 서비스를 이용 하실 때는 캐시된 콘텐츠가 없는 상태이므로 CDN 서버가 콘텐츠를 캐시 할 때 까지는 원본서버로 요청과 트래픽이 발생하여 시스템 부하가 발생 할 수 있습니다.

이런 상황으로 서비스의 품질이 저하되는 원인이 될 가능성이 있으므로 기존에 사용량이나 트래픽이 높은 경우 사이즈가 큰 콘텐츠(2GB 초과)를 사용하셨다면 서비스 신청 이전에 고객센터로 문의해 주시기 바랍니다.

기존에 CDN 서비스를 이용하지 않았던 고객으로서 처음 CDN 서비스를 이용하시는 고객이라면 트래픽과 부하에 대해서는 크게 문제 되지 않습니다.

CDN 서비스는 기존에 CDN 서비스를 이용하지 않았을 때 보다 많은 트래픽을 발생하거나 부하를 더 주는 경우가 없습니다.

## 5.2 ucloud CDN Standard FAQ

## ▣ Streaming 방식에 대한 제약은 무엇인가요?

- 원본 콘텐츠 서버로 ucloud storage의 파일박스만 사용이 가능하며, 파일박스 접근권한 설정을 공개로 설정하셔야 합니다.  
스트리밍 가능 파일 형식은 H.264/AAC Codec을 사용한 MP4 파일로 제한됩니다.

## ▣ 다른 CDN을 사용하다가 옮겼는데 속도가 너무 느립니다.

- 기존에 사용량이나 트래픽이 높은 경우 사이즈가 큰 콘텐츠(2GB 초과)를 사용하셨다면, 캐시된 콘텐츠가 없는 상태이므로 CDN 서버가 콘텐츠를 캐시 할 때 까지는 원본서버로 요청과 트래픽이 발생하여 시스템 부하가 발생 할 수 있습니다.  
이런 경우 사전에 지원이 가능하오니, 서비스 신청 이전에 고객센터로 문의해 주시기 바랍니다

## ▣ CDN을 사용하는데도 사용하지 않는 것과 비슷하게 부하가 큼니다.

- 콘텐츠 만료기간을 확인하시기 바랍니다.  
ucloud CDN 서비스를 담당하는 웹 캐시서버는 원본 콘텐츠가 갱신되었는지 여부를 1일마다 확인하도록 되어 있습니다.  
이 갱신주기 보다 짧거나 길게 설정을 하셔야 할 때는 웹 서버에서 콘텐츠 타입별 또는 디렉토리별로 Cache-Control 또는 Expires와 같은 설정을 하시면 관리가 용이합니다.

CDN 서비스에 제공되는 웹 캐시서버는 원본서버에서 응답하는 콘텐츠의 만료기간을 더 우선으로 하고 있으므로 원본 서버의 설정을 따르게 됩니다.

갱신주기를 짧게 설정하는 경우 캐시서버가 원본서버로 갱신여부를 자주 질의하게 되므로 CDN 서비스를 이용하는 효과가 낮아 질 수 있어, 웹 서버 설정시 충분한 고려가 필요합니다.

## ▣ CDN 캐시서버가 원본 콘텐츠 서버(Origin Server)에 접속하지 못합니다.

- 방화벽을 사용하는 고객이라면 방화벽에서 CDN 캐시서버 IP를 ACL에 허용해 주셔야 합니다.  
이 절차가 빠진 상태에서 서비스를 개통하게 되면 방화벽에서 캐시서버의 요청을 차단해 서비스가 단절되는 문제가 발생할 수 있습니다.

IPS 및 웹 방화벽 등 웹 서비스에 영향을 미칠 수 있는 보안 장비에 대해서도 화이트 리스트 등록 또는 ACL 허용이 필요합니다.

간혹, 캐시서비스를 이용하는 시작시기에는 방화벽에서 캐시서버 IP가 허용되도록 설정을 하였지만 ACL의 정책 만료기간이나 방화벽 특성에 따른 탐지오류로 인해 다시 차단되는 경우가 발생할 경우도 있으므로

이러한 우려가 없도록 ACL 허용 설정에 대해 정확한 룰셋 적용을 해 주셔야 합니다.

캐시서버가 웹 서버로 문제없이 연결될 수 있어야 안정적인 웹 캐시서비스를 이용하실 수 있습니다.

## 5.3 ucloud CDN Standard 이용방법

### 5.3.1 ucloud CDN Standard 서비스 관리

kt cloud biz 의 모든 서비스는 포탈 내 클라우드 콘솔을 통해 관리되고 있습니다.(신청, 변경, 정지, 해지, 조회 등)

ucloud CDN Standard 서비스는 [ucloud CDN] 메뉴를 통해 제어가 가능합니다.

서비스 유형	서비스명	상태	서비스 도메인
<input type="checkbox"/> 스트리밍	Test20151019	● 사용	-
<input type="checkbox"/> 다운로드	uc3890	● 정지	서비스 도메인
<input type="checkbox"/> 다운로드	uc3891	● 사용	서비스 도메인
<input type="checkbox"/> 다운로드	yk3890	● 사용	서비스 도메인
<input type="checkbox"/> 다운로드	testtest0120	● 사용	test0120test0120
<input type="checkbox"/> 다운로드	yk3891	● 사용	서비스 도메인
<input type="checkbox"/> 다운로드	tw3892	● 정지	서비스 도메인
<input type="checkbox"/> 다운로드	tw3895	● 사용	서비스 도메인
<input type="checkbox"/> 다운로드	tw399R	● 사용	서비스 도메인

- CDN 서비스 리스트: ucloud CDN Standard 서비스를 조회하고, 신청/변경/정지/해지하실 수 있습니다.
- 통계: ucloud CDN Standard 서비스의 실시간 사용량 뿐만 아니라 최근 1년간의 사용량을 조회하실 수 있습니다.
- API Key: OpenAPI를 사용하기 위한 Key를 조회하실 수 있습니다. ucloud server 메뉴의 API Key와 동일한 값입니다.

특정 서비스를 체크할 경우, 하단의 상세정보 탭에서 해당 서비스의 정보를 확인하실 수 있습니다. 각 항목에 대한 설명은 바로 아래의 [서비스 신청] 에서 확인하실 수 있습니다.

## ▣ 서비스 신청

### CDN 서비스 신청

UCloud CDN 서비스를 신청 할 수 있습니다.

서비스 유형:  다운로드  스트리밍

원본 컨텐츠 서버:

\* 영문, 숫자, 점(.), 콤마(,), 하이픈(-)만 입력 가능합니다.  
\* 원본파일이 존재하는 실제 서버의 IP주소 또는 도메인 주소를 입력해주세요.  
\* 콤마(,)로 연결하면 여러 개 입력도 가능합니다. 예) 123.123.123.123, origin.ktics.com

서비스명:

서비스 도메인:   CDN도메인을 서비스 도메인으로 사용

\* 영문, 숫자, 점(.), 콤마(,), 하이픈(-)만 입력 가능합니다.  
\* Multi-Domain은 지원하지 않습니다.

서비스 설명:

- 서비스 유형
  - 다운로드 서비스인지, 스트리밍 서비스인지 선택합니다.
  - 스트리밍 서비스는 원본 컨텐츠 서버로 ucloud storage만 지원합니다.
- 원본 컨텐츠 서버
  - CDN 서비스를 적용하기 위한 서버의 IP 주소 혹은 도메인을 입력합니다. 콤마(,)로 연결하여 여러 개 입력도 가능합니다.
  - ucloud storage의 파일박스를 원본 컨텐츠 서버로 사용하실 수 있습니다. 단, 이 경우에는 여러 개 동시 입력이 불가능합니다.
- 서비스명: CDN 서비스 이름을 입력합니다.
- 서비스 도메인
  - CDN 서비스를 제공하기 위한 도메인을 입력합니다. 서비스 도메인은 **Multi-Domain**을 지원하지 않습니다.

- CDN도메인을 서비스 도메인으로 사용 체크박스에 체크를 하면 ucloud CDN 제공 도메인을 이용하실 수 있습니다.

o 서비스 설명: 어떤 용도의 CDN 서비스인지 메모하기 위한 용도입니다. 입력하지 않으셔도 신청이 가능합니다.

### □ 서비스 정지 및 해지



o 서비스 정지: 사용 중인 CDN 서비스를 선택 후, Action > 정지 버튼을 클릭하여 서비스 정지가 가능합니다.  
- 서비스를 정지하여도 기본 요금이 발생하므로 주의해주시기 바랍니다.  
- 정지한 서비스는 Action > 시작 버튼을 클릭하여 언제든지 재시작이 가능합니다.

o 서비스 해지: 정지된 CDN 서비스를 선택 후, Action > 해지 버튼을 클릭하여 서비스 정지가 가능합니다.

## 5.3.2 Purge

### □ Purge란?

Purge는 CDN 캐시서버에 저장되어있는 콘텐츠를 삭제하는 툴(tool)이며, 원본서버의 콘텐츠가 변경되었을 때 즉각적으로 캐시서버에 반영하기 위한 방법으로 제공되고 있습니다.

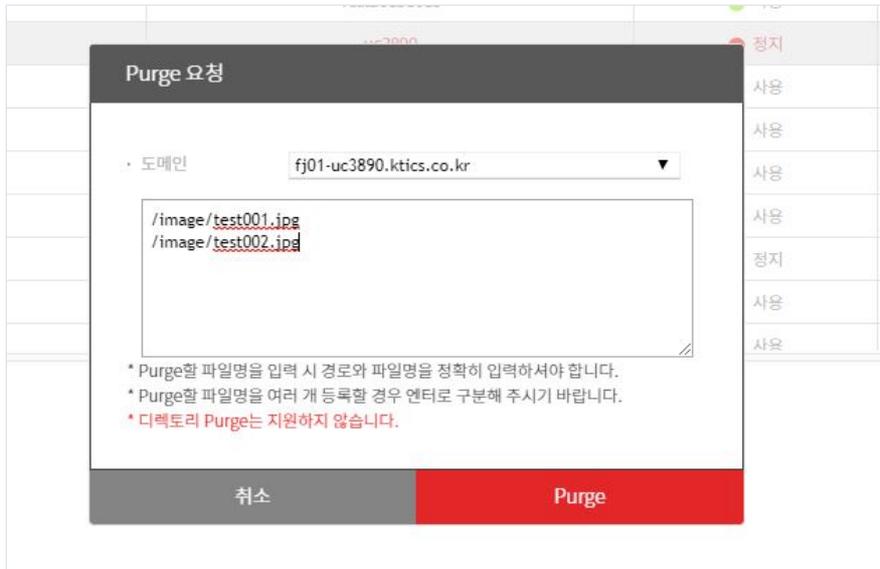
캐시서버는 사용자로부터 요청받은 콘텐츠가 저장되어 있지 않을 경우, 원본서버로부터 다운로드하여 다시 캐싱하도록 되어 있는데 이런 캐시서버의 특성을 이용하여 원본서버에서 변경된 콘텐츠가 캐시서버에 반영될 수 있도록 Purge를 사용하여 캐시서버의 기존 콘텐츠를 삭제합니다.

만약 원본서버의 콘텐츠를 변경하신 후에 Purge를 하지 않으면 캐시서버는 갱신주기에 의해서 원본서버의 콘텐츠가 변경되었음을 판단하기 전까지는 캐시서버에 저장된 콘텐츠를 계속 서비스 하게 됩니다. Purge는 CDN서비스를 하고 있는 캐시서버의 콘텐츠를 삭제하는 것이며 원본서버의 콘텐츠를 컨트롤 하지는 않습니다.

무분별한 Purge 사용은 원본서버의 부하로 이어질 수 있으므로 한번에 많은 Purge에는 주의를 요합니다.

### □ Purge 기능의 이용

- 1) CDN 서비스 선택 > Action > Purge 클릭
- 2) 도메인 선택: CNAME 대상 도메인 또는 실제 사용 중인 서비스 도메인 선택가능  
(xxx-서비스명.ktics.co.kr 도메인은 해당 도메인을 실제 서비스도메인으로 사용하는 경우에만 선택)



3) Purge 할 URI 입력: 파일명 입력 시 경로와 파일명을 정확히 입력해야 하며, 여러 개 등록 시에는 엔터로 구분  
 - 반드시 도메인을 제외한 경로를 입력해주세요. ex) /images/sample.jpg

4) Purge 실행

### □ Purge 주의사항

- Purge 사용은 원본서버의 부하로 이어질 수 있으므로 한번에 많은 양의 Purge 사용은 주의를 요합니다.
- 캐시서버에 저장된 콘텐츠를 사용자가 이용하고 있을 경우 Purge 수행이 불가합니다.
- Purge 수행이 불가 할 경우 원본서버의 콘텐츠 파일명을 변경하여 다시 업로드하는 방법을 권고 드립니다. (파일명 변경으로 인해 캐시서버에서 응답을 받지 못하므로 원본서버에서 콘텐츠를 찾아오기 때문)

## 5.3.3 ucloud storage 연동

ucloud CDN Standard 서비스의 원본 콘텐츠 서버로 ucloud storage 파일박스 선택이 가능합니다.

### □ 파일박스 접근권한

연동을 위해서는 먼저 다음과 같이 파일박스의 접근권한을 공개로 설정하여야 합니다.

파일박스명	파일수	사용량	접근권한 관리
catchingtest	0	0.00 KB	공개
con1	0	0.00 KB	비공개
encoder_test	0	0.00 KB	공개
yongyeon	1	20.50 KB	공개

접근권한 설정을 공개로 설정 시, 다음과 같이 원본 콘텐츠 서버에 파일박스가 노출됩니다.

CDN 서비스 정보수정 ☐ 온라인문의 ☐ 매뉴얼

서비스 유형  다운로드  스트리밍

원본 컨텐츠 서버

Availability Zone

파일박스

\* 원본파일을 저장한 파일박스를 선택해 주세요.  
 \* { ucloud storage에서 공개한 파일박스만 선택이 가능합니다 }  
 \* 스트리밍 가능 파일 형식 H.264/AAC Codec을 사용한 MP4 파일

서비스명  20byte

서비스 도메인   CDN도메인을 서비스 도메인으로 사용

\* 영문, 숫자, 점(.), 콤마(,), 하이픈(-)만 입력 가능합니다.

서비스 설명

4000byte

### ▣ Cloudberry 사용 시 주의사항

ucloud storage 서비스 이용을 위해 Cloudberry를 사용하실 경우 다음 사항에 대한 주의가 필요합니다.

Cloudberry를 사용하여 파일을 업로드 하는 경우에는, 의도하지 않아도 기본적으로 10MB 단위로 분할되어 업로드됩니다.

분할 크기 설정은 cloudberry의 option - Advanced - Use chunks 에서 가능하며, Chunk transparency 체크 박스를 해제하면 분할된 파일들을 볼 수 있습니다.

Cloudberry와 Swift API의 파일 분할 업로드 방식에 차이가 존재하여, Cloudberry에서 container에 대한 공개 권한 설정이 적용되지 않습니다.

때문에, API를 미용하여 ACL 설정이 필요합니다.

자세한 내용은 하기의 링크를 참고해주시기 바랍니다.

[https://ucloudbiz.kt.com/portal/portal.openapiinfo.html?menu=storage-storage\\_api\\_cloudberry\\_guide](https://ucloudbiz.kt.com/portal/portal.openapiinfo.html?menu=storage-storage_api_cloudberry_guide)

### 5.3.4 통계 확인

클라우드 콘솔 > ucloud CDN > 통계 메뉴에서 CDN 미터링 정보를 확인하실 수 있습니다. 과월 데이터는 최근 1년까지의 데이터만 조회가 가능합니다.

통계 종류(일별/시간대별), 서비스, 기간을 선택하신 후 검색 버튼을 클릭하시면 하단에 해당 기간동안의 총합 데이터와 상세 현황이 나타납니다.

통계 ☐ 온라인문의 ☐ 메뉴얼

ucloud CDN 상품 이용 현황을 보여줍니다.  
과월 데이터는 최근 1년 동안만 조회 가능합니다.

[다운로드](#) [스트리밍](#)

일별통계 ▼ 서비스 선택 sz6503 ▼ 기간 2018 ▼ 10 ▼ 검색

※ 시간대별 통계 요청 시 정보 조회까지 약 1-2분의 시간이 소요됩니다.

· 다운로드 현황

전송량	<b>0.000GB</b>	요청수	<b>270</b>
-----	----------------	-----	------------

· 현황 상세 엑셀저장

일자	다운로드 전송량	다운로드 요청수
16	125.53KB	270

○ 서비스 종류 선택

- 다운로드 서비스의 조회를 원하실 경우 다운로드 탭을, 스트리밍 서비스의 경우에는 스트리밍 탭을 선택합니다.

기본화면은 다운로드 서비스 통계 화면입니다.

○ 통계 종류

- 일별통계: 조회를 원하시는 년월을 선택하시고 검색하시면 현황 상세 리스트에 각 일자별 전송량/요청수가 나타납니다.

- 시간대별 통계: 조회를 원하시는 일자를 선택하시고 검색하시면 현황 상세 리스트에 각 시간대별 전송량/요청수가 나타납니다.

○ 서비스 선택

- 서비스명이 아닌 서비스 아이디를 선택하셔야 합니다.

서비스 아이디는 CDN 서비스 리스트 > 특정 서비스 선택 > 상세정보에서 확인이 가능합니다.

○ 기간 선택

- 조회를 원하시는 기간을 선택하신 후 반드시 검색 버튼을 클릭해야 합니다.

- 일별 조회는 년월을, 시간대별 조회는 일자까지 기간 선택이 필요합니다.

- 과월 데이터는 최근 1년까지의 데이터만 조회가 가능합니다.

## 5.4 ucloud CDN Standard 기능

ucloud CDN Standard 서비스를 이용할 때, 포탈 이용방법 외에 서비스를 잘 활용하기 위한 기능 측면의 가이드를 제공합니다.

목차는 다음과 같습니다.

### 5.4.1 스트리밍 서비스

### 5.4.2 캐시 기능

### 5.4.3 DNS 설정 방법

### 5.4.4 CDN 서비스 테스트 방법

### 5.4.1 스트리밍 서비스

이번 가이드에서는 스트리밍 서비스를 이용하기 위한 대표적인 프로토콜 2가지 이용 방법에 대해 소개합니다. 가이드 드리는 프로토콜은 iOS에서 사용하는 HLS 프로토콜과 Adobe Flash Player에서 사용하는 RTMP 프로토콜입니다.

#### ▣ HLS(HTTP Live Streaming) 프로토콜

Apple iPhone, iPad, iPod의 운영체제인 iOS에서 사용하는 표준 HTTP 기반 스트리밍 프로토콜입니다.

#### ○ 지원대상

- 디바이스 : iPhone/iPad/iPod
- 운영체제 : iOS 3.0 이상
- 콘텐츠형식 : MP4 (H264,AAC), MP3

#### ○ m3u8 파일

m3u8 파일은 HLS에 필요한 메타데이터를 담고 있는 파일이며, 따로 올릴 필요 없이 ucloud CDN 에서 자동 생성 됩니다.

#### ○ HLS 서비스 URL

##### 1) 사용법

```
[ uCloud CDN Prefix URL ] + [콘텐츠] + /playlist.m3u8
```

- MP4/M3에 대한 ucloud CDN Prefix URL: http://[CDN도메인]/vol-xxx6/\_definst\_
- HLS는 TCP 80포트 사용

##### 2) 사용 예

```
http://stm.ktics.co.kr/vol-xxx/_definst_/sample.mp4/playlist.m3u8
```

#### ○ iOS에서 재생하기 - A Tag를 이용한 재생

##### 1) 사용법

```
<HTML>  
<BODY>  
  <A HREF="[ uCloud CDN Prefix URL ] + [콘텐츠] + /playlist.m3u8">SAMPLE</A>  
</BODY>  
</HTML>
```

##### 2) 사용 예

```
<html>  
<body>  
<a href="http://stm.ktics.co.kr/vol-xxx/_definst_/sample.mp4/playlist.m3u8">sample</a>  
</body>
```

#### ○ iOS에서 재생하기 - Video/Audio Tag를 이용한 재생

Video/Audio Tag는 HTML5에 포함된 멀티미디어 콘텐츠를 웹페이지에 삽입하기 위한 Tag입니다. iOS의 Safari 브라우저는 Video/Audio Tag를 지원하고 있습니다.

##### 1) 사용법

```
<HTML>  
<BODY>  
  <VIDEO SRC="[ uCloud CDN Prefix URL ] + [콘텐츠] + /playlist.m3u8" controls>  
</VIDEO>  
</BODY>  
</HTML>
```

##### 2) 사용 예

```
<html>  
<body>  
<video src=" http://stm.ktics.co.kr/vol-xx x/_definst_/sample.mp4/playlist.m3u8" controls>
```

```
</video>
</body>
</html>
```

## ◦ 인증서비스 사용방법

인증서비스란? 클라이언트가 HTTP URL에 인증토큰을 첨부해서 접속할 때, ucloud CDN HLS 서버가 URL에 포함된 인증토큰을 검사해서 올바르게 서비스를 제공하고, 인증토큰이 올바르지 않으면 서비스를 제공하지 않는 서비스입니다.

## ◦ 인증서비스 URL 형식

### 1) 사용법

```
[HLS 스트리밍 URL] ? token = [인증토큰] & expr = [인증유효시간]
```

### 2) 사용 예

```
http://stm.ktics.co.kr/volxxx/_definst/_sam
ple.m p4/playlist.m 3u8?token=abcdef gh&expr=abcdef
```

## ◦ 인증 파라미터

- file : 스트리밍할 파일명입니다
- token : 인증토큰으로 32자리로 이루어집니다.
- expr : 인증토큰의 만료시간으로 unix timestamp의 hex 값입니다.  
\* unix timestamp는 세계표준시간 기준입니다.

## ◦ token/expr 생성방법 및 사용 예

### 1) 사용법

```
expr_rel = 60 // 인증 토큰 유효 시간 지정 (단위: 초)
secret = "123456789" // 서비스 생성 과정에서 설정한 인증암호
filename = "/sample.mp4" // /을 포함해서 파일명 설정
current = get_curr_unix_timestamp() // 현재 UNIX TIMESTAMP
expr = convert_int_to_hex(current + expr_rel) // 인증 토큰 유효 시간 (16진수 문자열)
message = concatenate(secret, filename, expr) // 인증 토큰 생성에 필요한 문자열 생성
token = md5(message) // MD5 Hash를 이용 인증 토큰 생성
```

### 2) PHP 사용 예

```
$expr_rel = 60;
$secret = '12345 678 90';
$filename = '/sample.m p4';
$current = time();
$expr = dechex($current + $expr_rel);
$message = $secret.$filename.$expr;
$token = md5($message);
```

## ◦ 시간동기화

인증 토큰을 발급하는 서버는 Unix/Linux는 kr.pool.ntp.org 시간서버와, Windows는 time.windows.com 과 시간 동기화가 되어 있어야 합니다.

동기화 방법은 다음과 같습니다.

- Unix/Linux 시간동기화: ntpdate 명령어 또는 ntp 서비스를 이용해서 주기적으로 동기화(OS별 설정은 OS제 조사에 문의 바랍니다.)

- Windows 시간동기화: 제어판 > 시계, 언어 및 국가별 옵션 > 날짜 및 시간 > 인터넷 시간에서 설정(자세한 동기화 방법은 Microsoft에 문의 바랍니다.)

## ▣ RTMP(Real Time Messaging Protocol)

Adobe Flash Player에서 사용하는 스트리밍 프로토콜입니다.

### ○ 지원대상

- Adobe Flash Player 10.x 이상이 설치되는 모든 디바이스(PC: Windows, Linux, Mac / Mobile: Android 2.2 이상)
- 콘텐츠형식 : MP4 (H264,AAC),MP3, FLV  
(디바이스 별 해상도, 화면크기, 코덱 설정은 Adobe 에 문의 바랍니다.)

### ○ RTMP 이용방법

RTMP 스트리밍 서비스를 이용하기 위해서는 Adobe Flash Player와는 별개로 JW Player와 같은동영상 플레이어가 필요합니다.

플레이어는 JW Player 또는 OSMF 와 같이 널리 알려진 플레이어를 사용하시거나, ActionScript를 이용 직접 개발해서 사용하실 수 있습니다.

### ○ RTMP URL

#### 1) 사용법

```
[ ucloud CDN Prefix URL ] + [콘텐츠]
```

- ucloud CDN Prefix URL은 rtmp://[CDN도메인]/vol-xxx/\_definst\_/
- RTMP는 TCP 1935포트 사용

#### 2) 사용 예

```
rtmp://stm.ktics.co.kr/vol-xxx/_definst_/sample.mp4
```

### ○ ActionScript RTMP 파라미터

위의 RTMP URL을 아래와 같이 2개로 나누어서 사용하시면 됩니다

#### 1) NetConnection.connect

```
rtmp://stm.dosirak.com/vol-xxx
```

#### 2) NetStream.play

```
sample.mp4
```

\* 보다 자세한 사항은

[http://help.adobe.com/en\\_US/FlashMediaServer/3.5\\_Deving/flashmediaserver\\_3.5\\_dev\\_guide.pdf](http://help.adobe.com/en_US/FlashMediaServer/3.5_Deving/flashmediaserver_3.5_dev_guide.pdf)을 참고하십시오.

### ○ JW Player 이용방법

JW Player 5의 경우, 위의 RTMP URL을 streamer와 file으로 나누어서 사용하시면 됩니다  
JW Player 6의 경우, 위의 RTMP URL을 그대로 file 파라미터로 사용합니다.

<JW Player 5에서 Object / Embed 문을 위한 JW Player flashvars 파라미터 설정>

#### 1) streamer 파라미터

streamer=rtmp://stm.dosirak.com/vol-xxx

## 2) file 파라미터

file=sample.mp4

(보다 자세한 사항은 <http://www.longtailvideo.com/support/jw5/31145/rtmp-streaming> 을 참고하세요.)

## 3) 예제

```
<object classid='clsid:D27CDB6E-AE6D-11cf-96B8-444553540000' width='300' height='300'  
  id='player1' name='player1'>  
  <param name='movie' value='player.swf'>  
  <param name='allowfullscreen' value='true'>  
  <param name='allowscriptaccess' value='always'>  
  <param name='flashvars'  
    value='streamer=rtmp://stm.dosirak.com/vol-xxx  
      &file=sample.mp4&type=rtmp'>  
  <embed id='player1' name='player1' src='player.swf' width='300' height='300'  
    allowscriptaccess='always'  
    allowfullscreen='true'  
    flashvars='streamer=rtmp://stm.dosirak.com/vol-xxx&file=sample.mp4&type=rtmp'  
  
  />  
</object>
```

<JW Player 6에서 Object / Embed 문을 위한 JW Player flashvars 파라미터 설정>

## 1) file 파라미터

file=rtmp://stm.dosirak.com/vol-xxx/\_definst\_/sample.mp4

(보다 자세한 사항은 <http://www.longtailvideo.com/support/jw-player/288354/usingrtmp-streaming> 을 참고하세요.)

## ◦ 인증서비스 사용방법

인증서비스란? 클라이언트가 RTMP URL에 인증토큰을 첨부해서 접속할 때, ucloud CDN RTMP 서버는 URL에 포함된 인증토큰을 검사해서 올바르면 서비스를 제공하고, 인증토큰이 올바르지 않으면 서비스를 제공하지 않는 서비스 입니다.

## ◦ 인증 URL 형식

### 1) 사용법

```
[RTMP 스트리밍 URL] ? token = [인증토큰] & expr = [인증유효시간]
```

### 2) 사용 예

```
rtmp://stm.ktics.co.kr/vol-xxx/_definst_/sample.mp4?token=abcdefg&expr=abcdef
```

## ◦ 인증서비스 사용을 위한 ActionSript RTMP 파라미터

### 1) NetConnection.connect

rtmp://stm.ktics.co.kr/vol-xxx ?token=1234567890abcdef&expr=12345678&file=sample.mp4

\* 인증서비스를 위해 token, expr, file 파라미터 추가

### 2) NetStream.play

sample.mp4 (인증서비스 사용 시에도 영향 없음)

## ◦ 인증서비스 사용을 위한 JW Player 5의 Object / Embed 문을 위한 JW Player flashvars 파라미터

### 1) streamer 파라미터

streamer=rtmp://stm.dosirak.com/vol-xxx?token=1234567890abcdef&expr=12345678&file=sample.mp4

\* 인증서비스를 위해 token, expr, 파라미터 추가

2) file 파라미터

file=sample.mp4 (인증서비스 사용 시에도 영향 없음)

### ◦ JW Player 6의 Object / Embed 문을 위한 JW Player flashvars 파라미터

1) file 파라미터

file=rtmp://stm.dosirak.com/vol-xxx/\_definst\_/sample.mp4?

token=1234567890abcdef&expr=12345678&file=sample.mp4

\* 인증서비스를 위해 token, expr, 파라미터 추가

### ◦ 인증 파라미터란?

token : 인증토큰으로 32자리로 이루어집니다.

expr : 인증토큰의 만료시간으로 unix timestamp의 hex 값입니다.

\* unix timestamp는 세계표준시간 기준입니다.

<token/expr 생성방법 및 사용 예>

1) 사용법

```
expr_rel = 60 // 인증 토큰 유효 시간 지정 (단위: 초)
secret = "1234567890" // 서비스 생성 과정에서 설정한 인증암호
filename = "/sample.mp4" // /을 포함해서 파일명 설정
current = get_curr_unix_timestamp() // 현재 UNIX TIMESTAMP
expr = convert_int_to_hex(current + expr_rel) // 인증 토큰 유효 시간 (16진수 문자열)
message = concatenate(secret, filename, expr) // 인증 토큰 생성에 필요한 문자열생성
token = md5(message) // MD5 Hash를 이용 인증 토큰 생성
```

2) PHP 사용 예

```
$expr_rel = 60;
$secret = '1234567890';
$filename = '/sample.mp4';
$current = time();
$expr = dechex(time()+$expr_rel);
$messages = $secret.$filename.$expr;
$token = md5($messages);<
```

### ◦ 시간동기화

인증 토큰을 발급하는 서버는 Unix/Linux는 kr.pool.ntp.org 시간서버와, Windows는 time.windows.com 과 시간 동기화가 되어 있어야 합니다.

동기화 방법은 다음과 같습니다.

- Unix/Linux 시간동기화: ntpdate 명령어 또는 ntp 서비스를 이용해서 주기적으로 동기화(OS별 설정은 OS제 조사에 문의 바랍니다.)

- Windows 시간동기화: 제어판 > 시계, 언어 및 국가별 옵션 > 날짜 및 시간 > 인터넷 시간에서 설정(자세한 동기화 방법은 Microsoft에 문의 바랍니다.)

## 5.4.2 캐시 기능

### ▣ 캐시에 대한 오해와 이해

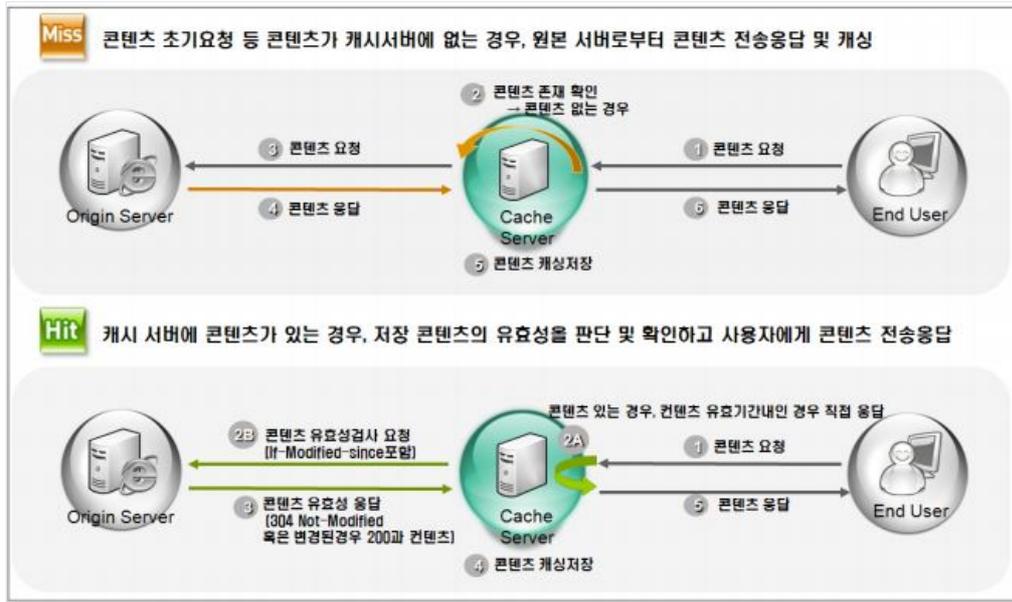
#### ◦ HIT과 MISS의 의미

캐시서버는 사용자가 요청하는 콘텐츠가 자신의 Memory나 HDD에 저장(cache)되어 있는지를 먼저 확인합니다.

- HIT: 이 때 요청한 콘텐츠가 캐시서버에 저장되어 있다고 판단을 할 경우,

원본서버로 요청을 하지 않고 캐시서버에 저장된 콘텐츠를 사용자에게 전송하며 'HIT' 이라고 응답을 합니다.

- MISS: 그리고 요청한 콘텐츠가 캐시서버에 저장되어 있지 않다고 판단을 하면 원본서버로 요청을 하게 되는데 이때 캐시서버는 'MISS'라고 응답을 합니다.



### o Cache에 대한 이해

'MISS'의 경우는 이전에 요청된 이력이 없는 경우 처음에만 MISS가 발생하며 이후부터는 HIT이 발생합니다. 그러나 아래와 같이 캐시를 할 수 없는 경우도 있습니다.

#### <캐시를 할 수 없는 경우>

- 1) 요청하는 METHOD가 POST 인 경우
  - ucloud CDN 서비스의 웹 캐시서버는 GET, HEAD의 Method만 캐시를 하고, POST Method는 캐시를 하지 않습니다.
- 2) 원본서버에서 응답헤더에 Last-Modified가 없는 경우
  - 웹 캐시서버는 원본서버에 갱신 여부를 확인하기 위해서 HTTP header의 Last-Modified 시간을 비교하므로 원본서버의 응답 헤더에 Last-Modified가 없으면 캐시를 하지 않습니다.
- 3) 캐시서버에 저장된 콘텐츠의 Last-Modified 시간이 원본서버의 콘텐츠 보다 최근일 때
  - 응답 헤더에 Last-Modified가 있더라도 원본서버에서 기존의 콘텐츠보다 더 오래된 콘텐츠로 변경한 경우에도 갱신을 하지 않고 기존의 저장된 콘텐츠를 전송합니다.
- 4) 요청하는 HTTP 헤더에 no-cache, no-store, max-age=0 등을 포함 하고 있는 경우
  - 원본서버의 Cache-Control과 Pragma 설정이 되어 있는 경우 캐시서버는 **자신의 설정보다 웹 서버의 설정을 우선으로 적용**하게 됩니다.
  - 그래서 no-cache와 같은 요청이 있을 경우 캐시서버에 저장되어 있는 콘텐츠를 사용하지 않고 'MISS'를 발생하게 됩니다.
- 5) 원본서버에서 응답하는 상태코드가 200 OK가 아닌 경우
  - 웹 캐시서버는 원본서버로부터 정상적인 응답을 받지 않은 경우 저장을 하지 않으나 404 Not Found의 경우는 사용자들의 악의적인 요청이 빈번히 발생 할 수 있어 웹 캐시서버에서 30초간 원본서버로 질의 없이 사용자의 동일한 요청에 대해서 404 Not Found를 응답합니다. 이를 'NEGATIVE HIT'이라고 합니다.

캐시를 할 수 없는 경우의 요청이나 응답을 받은 경우에는 콘텐츠를 저장하지 않으며 캐시서버는 프록시(proxy) 역할만 합니다.

캐시서버는 요청받은 콘텐츠가 저장되어 있더라도 1일이 지난 경우에는 해당 콘텐츠가 변경이 되었는지 여부를 확인하기 위해서 원본서버에 IMS(If-Modified-Since) 요청을 하여 변경 여부를 확인을 합니다.

이때 원본서버가 변경되지 않았다고 304 Not Modified 응답을 하면(이때 응답헤더만 받게 됨) 캐시서버는 저장된 콘텐츠를 사용자에게 전송하게 됩니다.

그리고 원본서버에서 변경이 된 경우에는 해당 콘텐츠를 원본서버로부터 콘텐츠를 다운로드 하게 되고 다운로드가 완료되면 원본서버에서 200 OK를 응답하게 되며 캐시서버는 내려 받은 최신 콘텐츠를 사용자에게 전송하게 됩니다.

만약 콘텐츠가 원본서버에서 삭제 된 경우에는 원본서버가 404 Not Found 라고 응답을 하고 캐시서버는 기존에 저장하고 있던 콘텐츠를 삭제하고 사용자에게 404 Not Found를 응답하게 됩니다.

웹 캐시서버와 원본서버가 네트워크가 단절되는 경우,  
또는 원본서버의 시스템 다운과 같이 웹 캐시서버가 원본서버로 갱신여부를 확인 할 수 없는 경우,  
캐시서버는 저장하고 있는 콘텐츠를 지속적으로 서비스 하게 됩니다.  
물론 저장되어 있지 않은 콘텐츠의 요청에 대해서는 원본서버로부터 받은 HTTP 에러 코드를 보여주게 됩니다.

## □ 효율적인 캐시서비스 방법

### ○ 효율적인 캐시정책의 기본

효율적인 캐시서비스를 위해 콘텐츠의 종류와 특성별로 분리를 하여 적절하게 캐시서비스를 이용하시는 것이 좋습니다.  
아래에 소개되는 효율적인 캐시 정책을 활용 하신다면 CDN 서비스의 빠른 응답시간과 높은 Hit율을 보장할 수 있습니다.

'HIT', 'MISS' 콘텐츠를 분류해 도메인으로 분리합니다.

캐시서버에서 'HIT' 콘텐츠와 'MISS' 콘텐츠를 함께 사용하도록 한 도메인을 사용하게 되면 'HIT' 되는 콘텐츠는 캐시서버에서 바로 응답을 하여 빠른 전송을 할 수 있지만 그렇지 않은 'MISS' 되는 콘텐츠는 캐시서버를 거쳐 원본서버에서 응답을 하게 되므로 그만큼 응답 시간이 지연 됩니다.

그래서 이런 경우는 도메인을 각각 분리하여 'HIT' 되는 콘텐츠의 도메인은 CDN 서비스를 이용하도록 설정을 하고 그렇지 않은 도메인은 웹 서버에서 직접 전송토록 하는 것이 효율적입니다.

### ○ 웹 서버에서 콘텐츠 만료기간 관리

ucloud CDN 서비스를 담당하는 웹 캐시서버는 원본 콘텐츠가 갱신되었는지 여부를 1일마다 확인하도록 되어 있습니다.

이 갱신주기 보다 짧거나 길게 설정을 하셔야 할 때는 웹 서버에서 콘텐츠 타입별 또는 디렉토리별로 Cache-Control 또는 Expires와 같은 설정을 하시면 관리가 용이합니다.

CDN 서비스에 제공되는 웹 캐시서버는 원본서버에서 응답하는 콘텐츠의 만료기간을 더 우선으로 하고 있으므로 원본 서버의 설정을 다르게 됩니다.

갱신주기를 짧게 설정하는 경우 캐시서버가 원본서버로 갱신여부를 자주 질의하게 되므로 CDN 서비스를 이용하는 효과가 낮아 질 수 있으므로 웹 서버 설정시 충분한 고려가 필요합니다.

### ○ 웹 서버 설정

웹 캐시서버의 동작원리는 원본서버로부터 새로운 콘텐츠를 받아 저장(cache)하고, 그 콘텐츠에 대한 갱신주기를 판단해 요청 받은 콘텐츠를 적절하게 전송하는 것입니다.

적절한 갱신주기 설정은 효율적인 캐싱을 위해 중요한 요소입니다. 콘텐츠에 대한 갱신정책은 웹 서버에서 설정 할 수 있으며, 캐시서버에서도 해당 콘텐츠에 대한 유효성 여부를 검사할 수 있는 항목을 제공하고 있지만 웹 서버의 설정을 우선으로 합니다.

[표 1]에서 보는 내용들은 일반적으로 웹 페이지 구성 시 캐시와 관련된 것이며, 이외에도 다양한 항목들이 콘텐츠 유효성 검사를 위한 캐시정책에 영향을 미치게 됩니다.

[표 1] 캐싱 콘텐츠 유효성 관련 항목

항목	내용	비고
마지막 변경 사항	해당 콘텐츠의 마지막 변경 시간을 기록	GMT 기준

(Last-Modified)		
E-Tag	서버에서 생성되는 유일한 인자로 콘텐츠의 중복성과 캐싱 유효성 여부를 확인할 수 있는 항목	HTTP 1.1
캐시컨트롤 (Cache-Control)	해당 콘텐츠에 대한 캐싱 관련 정책을 제어하는 항목 (max-age, no-cache, must-revalidate 등이 있음)	HTTP 1.1
Age	해당 콘텐츠가 갱신된 시점을 기준으로 초기화되며 새롭게 갱신되는 시점까지 1초 단위로 증가함	
Expires	해당 콘텐츠가 얼마동안 유효한지를 표시해주는 항목으로 명시된 시간 이후 캐싱 유효성 재확인	GMT 기준

웹 서버의 특성에 따라 설정을 수정하실 때는 캐시정책과 관련 있는지 여부를 확인하셔야 합니다. 웹 서버의 설정은 캐시서버의 캐시정책에도 영향을 미치게 되므로 설정 시에 유념하시고 변경 시 의문점이 있으면 캐시서비스 담당자에게 문의를 하시기 바랍니다.

### o 캐시 컨트롤 HTTP 헤더

HTTP 1.1 에서는 Cache-control Response Header라는 새로운 헤더 클래스를 지원합니다. 이는 만료시간 설정에 대한 문제 해결책으로 웹 제작자가 직접 콘텐츠를 조절할 수 있게 해줍니다. 캐시 컨트롤 헤더에 포함되는 옵션은 아래와 같습니다.

- o max-age=[초] : 객체가 업데이트가 필요 없다고 여길 수 있는 최대 시간을 정해줍니다. 만료시간과 마찬가지로 특정시간이라기 보다는 요청시간과 관련이 있습니다. 요청받은 시간으로부터 설정해준 [초]까지 콘텐츠가 업데이트 필요없는 객체라고 나타내줍니다.
- o s-max-age=[초] : max-age와 비슷합니다.
- o public : 캐시할 수 있는 인증된 요청임을 표시하는 것으로 일반적으로 HTTP 인증이 요구되는 경우는 자동적으로 응답은 캐시하지 않습니다.
- o no-cache : 매회 캐시된 객체를 사용자에게 전달하기 전에 캐시에서 원본서버로 객체를 요청합니다. 인증이 중요시 되는 경우 유용하며, 계속적으로 업데이트되어야 할 객체일 경우 유용합니다.
- o no-store : 어떠한 경우라도 객체가 캐시되지 않도록 합니다.
- o must-revalidate : 웹 캐시서버가 콘텐츠의 업데이트 여부에 대한 정보를 반드시 따르도록 정해줍니다. HTTP가 업데이트 되지 않은 콘텐츠에 대해서 특정 조건 하에서 사용자에게 제공하도록 합니다.

### o Freshness (신선도)

'Freshness(신선도)'란 클라이언트의 요청이 'HIT' 되었을 때 캐시서버에서 해당 콘텐츠를 직접 전달해도 되는지 판단하는 것을 말합니다.

이 때 'Fresh(신선한)'와 'Stale(신선하지 않은)'이라는 용어도 함께 사용되는데 'Freshness'는 캐시나 웹에서 아주 중요합니다.

캐시서버에서 신선도를 판단하지 않는다면 원본서버에서 콘텐츠가 변경되었음에도 불구하고 캐시서버는 이전에 캐시된 콘텐츠를 계속해서 응답하게 되므로 사용자는 새로운 콘텐츠를 받을 수 없게 됩니다.

캐시서버가 확인하는 것은 응답했던 콘텐츠의 유효기간(Expiration time)을 제일 먼저 확인하는데 이 정보는 이전에 HTTP 응답 헤더(HTTP response header)의 max-age나 Expires로 판단합니다.

예를 들어 이전에 'Cache-control : max-age=86400' (24시간) 라는 값을 가지고 있었는데 아직 응답한지 24시간이 지나지 않았으면 이 콘텐츠는 신선(Fresh)하다고 판단하여 원본서버에 요청하지 않고 캐시서버에서 즉시 응답해 주고 만약에 이 콘텐츠가 유효기간이 지났다면 원본서버로 유효한지를 확인하는 과정을 거치게 됩니다.

### o Validation (유효성)

캐시서버의 'HIT' 콘텐츠가 신선하지 않은 콘텐츠로 판단되면 원본서버로 재사용여부를 묻게 됩니다. 이 때 일반적으로 사용하는 부분이 마지막 수정시간(Last-Modified)이고 캐시가 원본서버로 보내는 헤더(HTTP Header)에 다음 [표 2]와 같은 내용을 포함하게 됩니다.

[표 2] 캐싱 콘텐츠 유효성 확인을 위한 헤더 정보

```
GET http://www.abc.com/image.gif HTTP/1.1  
  
If-Modified-Since: Sun, 8 Sep 2009 19:43:31 GMT
```

캐시의 If-Modified-Since 헤더를 받은 원본서버는 자신이 가지고 있는 콘텐츠와 비교하여 유효한지에 대해서 판단하게 됩니다.

그리고 사용 가능한 콘텐츠일 경우 캐시에게 콘텐츠를 전송하는 것이 아니라 '304 Not Modified' 라는 간단한 메시지만 보냅니다.

이 응답을 받은 캐시서버는 콘텐츠가 아직 유효하다고 결정, 캐시에 있는 새로운 콘텐츠에 대한 정보(Date, Expires)만 업데이트하게 되고

캐시에 있는 콘텐츠를 클라이언트에게 전송합니다.

만약 304가 아닌 응답 코드를 받았을 경우에는 실제 원본서버로부터 새 콘텐츠를 내려받고 자신의 콘텐츠와 교체한 다음 클라이언트에게 전송합니다.

콘텐츠 재사용이 유효한지 판단할 때는 If-Modified-Since 외에도 ETag 도 있는데 ETag는 서버가 응답할 때 자신의 메타정보를 가공해서 보낸 문자열로서,

캐시는 단지 ETag가 있을 경우에만 요청헤더에 ETag 문자열을 포함해서 보냅니다.

이 때 요청 받은 서버는 단순히 요청헤더에 포함된 ETag의 문자열 비교를 통해서 유효성(Validation)을 판단합니다.

### 5.4.3 DNS 설정 방법

ucloud CDN 서비스를 사용하기 위해서는 기존에 사용하시던 서비스 도메인을 CDN 도메인으로 연결 되도록 설정하는 것이 필요합니다.

DNS의 솔루션은 현재 여러가지가 사용되고 있지만 본문 에서는 가장 흔히 사용되는 DNS 솔루션인 BIND를 위주로 설명합니다.

DNS의 설정 미숙으로 인한 장애가 발생 할 수 있으니 주의가 필요합니다.

#### ▣ CNAME 레코드의 의미

CNAME 레코드는 "Canonical NAME"의 약어로서, 특정 도메인에 대한 서브도메인(sub domain)을 사용하기 위한 레코드입니다.

가장 많이 사용하는 A 레코드는 특정 IP에 대해서 서브도메인을 사용하는 방법이지만

CNAME은 IP가 아니라 도메인 또는 동일 zone 파일내의 다른 호스트를 사용하기 위한 방법 이라는 점에서 '별칭' 또는 '별명' 이라는 말로 쓰이기도 합니다.

BIND가 설치된 시스템에서는 아래 [표 5]와 같이 zone 파일을 수정합니다.

아래의 [표 1]은 abc.com 의 도메인에서 www 호스트를 CDN 도메인 'ab01-ab0123.ktics.co.kr' 로 CNAME 레코드를 이용하여 설정한 것입니다.

이 설정이 적용되면 ucloud CDN 서비스로 트래픽이 발생되고 CDN 서비스가 시작됩니다.

CNAME 으로 설정할 CDN 도메인은 CDN 신청하는 웹 페이지에서 'CDN 서비스 신청' 란에서 정보 입력후에 확인 가능합니다.

[표 1] CNAME 레코드 설정 예

```
$TTL 60  
IN NS ns.abc.com.  
ns IN A 123.123.123.123  
www IN CNAME ab01-ab0123.ktics.co.kr  
abc.com. IN A 123.123.123.123
```

#### TIP!

TTL 값이 큰 경우에는 CNAME 변경 후에도 TTL 시간만큼 CDN으로 서비스가 전환되는 시간이 길어지게 됩니다. TTL 시간을 고려하여 **CNAME 설정 전에 TTL을 줄여 놓고 설정**을 하시면 서비스 전환이 빠르며,

잘못된 설정으로 인한 장애 시의 복구 시간도 TTL 만큼 짧아지게 됩니다.  
적용 후 정상적인 서비스가 된다면 TTL을 이전과 동일하게 설정하시면 됩니다.

### □ DNS 설정 확인 방법

수정한 zone 파일의 설정이 DNS 동작에 문제가 없는지 확인합니다.

[표 2] zone file syntax 테스트 방법

```
# named-checkzone abc.com /var/named/abc_com.zonezone localhost/IN: loaded serial 420K
```

zone 파일의 syntax 테스트 결과에 문제가 없다면 수정한 zone 파일을 적용한 후,  
dig 명령([표 3] 참조)이나 nslookup 명령을 이용하여 수정된 CNAME 설정이 문제가 없는지 확인을 합니다.

[표 3] 네임서버 설정 확인 예

```
$ dig www.abc.com; <<>> DiG 9.2.4 <<>> www.abc.com;; global options: printcmd;; Got answer;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 22958;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 2, ADDITIONAL: 2;; QUESTION SECTION:;www.abc.com. IN A;; ANSWER SECTION:www.abc.com. 10478 IN CNAME ab01-ab0123.ktics.co.kr.ab01-ab0123.ktics.co.kr.60 IN CNAME cdn-ucloud2.ktics.co.kr.cdn-ucloud2.ktics.co.kr.0 IN A 234.234.234.234cdn-ucloud2.ktics.co.kr.0 IN A 234.234.234.235cdn-ucloud2.ktics.co.kr.0 IN A 234.234.234.236;; AUTHORITY SECTION:bd-01.ktics.co.kr. 1705 IN NS ns1.bd-01.ktics.co.kr.;; Query time: 6 msec;; SERVER: 168.126.63.1#53(168.126.63.1);; WHEN: Thu Dec 3 12:13:36 2009;; MSG SIZE rcvd: 201
```

## 5.4.4 CDN 서비스 테스트 방법

CDN 서비스가 정상적으로 서비스 되고 있는지 유무를 파악하기 위해서는 다음과 같은 방법을 이용하실 수 있습니다.

- 리눅스 계열: curl 혹은 wget을 이용
- 윈도우: 웹 브라우저에서 직접 url을 입력

본 내용에서는 리눅스 계열의 명령어를 이용하여 테스트하는 방법에 대해 기술하고 있습니다.

### □ wget을 이용한 방법

wget을 이용하여 'http://www.abc.com/image/13.jpg' 파일을 요청하는 기본적인 방법을 [표 1]에서 예를 들었습니다.

여러 대의 캐시서버 중 특정 캐시서버에 요청을 할때는 IP로 요청할 수 있지만, 이때는 Host 정보에 정확한 도메인이 있어야 합니다.

[표 1] wget을 이용한 CDN 서비스 테스트 예

```
$ wget -S http://www.abc.com/image/13.jpg  
$ wget -S --header='Host: www.abc.com' http://123.123.123.123/image/13.jpg
```

[표 2] wget 테스트 결과 예

```
--13:00:41-- http://123.123.123.123/image/13.jpg  
=> `0.jpg'  
Connecting to 123.123.123.123:80... connected.  
HTTP request sent, awaiting response...  
HTTP/1.0 200 OK  
Date: Thu, 15 Dec 2011 04:00:30 GMT  
Expires: Sat, 14 Jan 2012 04:00:30 GMT
```

```
Server: Apache/2.2.3 (CentOS)
Last-Modified: Fri, 24 Sep 2010 16:50:13 GMT
ETag: "288800c-400-491042c1f7b40"
Accept-Ranges: bytes
Content-Length: 1024
Cache-Control: max-age=2592000
Content-Type: image/jpeg
X-Cache: HIT from cache.ktics.co.kr
X-Cache: MISS from i1.cache.ktics.co.kr
Connection: keep-alive
Length: 1,024 (1.0K) [image/jpeg]
```

응답 헤더 정보에서 200 OK 상태코드를 받아 정상적으로 서비스가 되고 있음을 확인 하였다면 정상적으로 서비스가 되고 있음을 판단 할 수 있습니다.

[표 3] 은 '400 Bad Request'가 발생한 결과의 예시입니다.

이는 CDN 서비스를 하는 웹 캐시 서버 또는 원본서버에서 잘못된 요청을 받았다고 판단 하였을 경우에 나타나며 서비스가 되지 않는 상황입니다.

주로 잘못된 도메인 설정으로 인해서 발생하는 경우가 많은데 아래와 같은 확인을 하시기 바랍니다.

- wget 사용시 도메인 또는 Host 정보를 정확하게 사용하였는지 확인하십시오.
- dig를 이용하여 도메인 룩업시 wget에서 사용한 IP가 있는지 확인하십시오.
- wget을 이용하여 웹 서버로 요청 하여 상태코드를 확인하십시오.
  - 웹서버로 요청시 정상적이라면 ucloud 고객센터로 문의해 주시기 바랍니다.
  - 웹서버에서도 동일한 응답 코드를 받았다면 웹서버의 도메인 설정을 확인 하십시오.

[표 3] 비정상적인 응답헤더 정보 예

```
--11:57:35-- http://123.123.123.123/500K/13.jpg
=> `13.jpg'
Connecting to 123.123.123.123:80... connected.
HTTP request sent, awaiting response...
HTTP/1.0 400 Bad Request
Server: ics-cache/3.5.1
Date: Thu, 03 Dec 2009 02:57:35 GMT
Content-Type: text/html
Content-Length: 1312
Expires: Thu, 03 Dec 2009 02:57:35 GMT
X-Cache-Error: ERR_INVALID_REQ 0
X-Cache: MISS from i2.cache.ktics.co.kr
Proxy-Connection: close
11:57:35 ERROR 400: Bad Request
```

응답헤더는 상황에 따라 다양하게 나타날 수 있으며 응답코드 또한 결과에 따라 다양한 코드로 표기 됩니다. 이에 대해 대표적인 응답헤더와 응답코드를 사전에 파악해 놓으시면 관리에 편리합니다. 응답코드 정보는 인터넷에서 'HTTP response code'를 검색하시면 상세한 내용을 보실 수 있습니다.

## □ curl을 이용하는 방법

curl을 사용하는 방법도 wget과 옵션의 사용법이 다를 뿐 크게는 차이는 없습니다.

어느 툴을 사용하느냐에 따라서 결과가 달라지는것은 아니지만 보여주는 내용에는 다소 차이가 있으므로 사용이 편리한 툴을 선택하셔서 사용하시면 됩니다.

[표 4] curl을 이용한 CDN 서비스 테스트 예

```
$ curl -v http://www.abc.com/image/13.jpg
$ curl -v -H 'Host: www.abc.com' http://123.123.123.123/image/13.jpg
```

[표 5] curl 테스트 결과 예

```
* About to connect() to 123.123.123.123 port 80
* Trying 123.123.123.123... * connected
* Connected to 123.123.123.123 (123.123.123.123) port 80
> GET /image/13.jpg HTTP/1.1

User-Agent: curl/7.12.1 (i686-redhat-linux-gnu) libcurl/7.12.1 OpenSSL/0.9.7a zlib/1.2.1.2
libidn/0.5.6

Pragma: no-cache

Accept: */*

Host: image.test.com

< HTTP/1.0 200 OK
< Date: Thu, 15 Dec 2011 04:00:30 GMT
< Server: Apache/2.2.3 (CentOS)
< Last-Modified: Fri, 24 Sep 2010 16:50:13 GMT
< ETag: "288800c-400-491042c1f7b40"
< Accept-Ranges: bytes
< Content-Length: 1024
< Cache-Control: max-age=2592000
< Expires: Sat, 14 Jan 2012 03:55:07 GMT
< Content-Type: image/jpeg
< X-Cache: MISS from cache.ktics.co.kr
< X-Cache: MISS from i2.cache.ktics.co.kr
< Connection: close

* Closing connection #0
```