Amazon CloudFront

개발자 안내서

API 버전 2016-09-29

Amazon CloudFront: 개발자 안내서

Copyright © 2020 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Amazon CloudFront란? 1
- 콘텐츠를 전송하도록 CloudFront를 설정하는 방법 1
- 사용 사례 3
  - 정적 웹 사이트 콘텐츠 전송 속도 향상 3
  - 온디맨드 비디오 또는 라이브 스트리밍 비디오 제공 3
  - 시스템 처리를 통해 특정 필드 암호화 4
  - 엣지에서의 사용자 지정 4
  - Lambda@Edge 사용자 지정을 사용하여 프라이빗 콘텐츠 제공 4
- CloudFront에서 콘텐츠를 제공하는 방법 4
  - CloudFront에서 사용자에게 콘텐츠를 제공하는 방법 5
  - CloudFront에서 리전 엣지 캐시를 사용하는 방식 5
- CloudFront 엣지 서버의 위치 및 IP 주소 범위 6
- CloudFront에 액세스 7
- 처음으로 Amazon CloudFront 사용하기 7
- AWS Identity and Access Management 7
- CloudFront 요금 8
  - CloudFront 배포의 가격 등급 선택 9

설정 10
- AWS에 가입 10
- 계정에 액세스 10
  - 콘솔에 액세스 11
  - API, AWS CLI, Windows PowerShell용 AWS 도구 또는 AWS SDK에 액세스 11
- IAM 사용자 생성 11
- AWS Command Line Interface 또는 Windows PowerShell용 AWS 도구 설정 13
- AWS SDK 다운로드 13

시작하기 14
- 간단한 배포로 시작하기 14
  - 사전 조건 14
  - 1단계: Amazon S3에 콘텐츠 업로드 및 객체 권한 부여 14
  - 2단계: CloudFront 배포 만들기 16
  - 3단계: 링크 테스트 21
- AWS for WordPress 시작하기 22
  - 사전 조건 22
  - 1단계: 플러그인 설치 24
  - 2단계: 플러그인으로 CloudFront 구성 및 사용 25
  - (선택 사항) 사이트 가속화 비활성화 27
  - (선택 사항) 사이트 가속화 제거 및 CloudFront 배포 삭제 27
  - (선택 사항) 플러그인 비활성화 및 제거 28
  - (선택 사항) Amazon Polly 콘텐츠용 CloudFront 배포 만들기 28
  - 문제 해결 29

배포 작업 31
- 배포의 개요 31
  - 배포 시 사용할 수 있는 작업 32
  - 배포 생성 및 배포 업데이트에 필요한 필드 32
- 배포 만들기, 업데이트, 삭제 33
  - 배포 만들기 단계 34
  - 배포 만들기 35
  - 지정하는 값 35
  - 표시되는 값 58
  - 배포 테스트 59
  - 배포 업데이트 59
  - 배포 삭제 60
- 여러 오리진 사용 61
  - Amazon S3 버킷을 오리진으로 사용 62

오리진에 대한 웹 사이트 엔드포인트로 구성된 Amazon S3 버킷 사용 62
오리진에 대한 MediaStore 컨테이너 또는 MediaPackage 채널 사용 63
Amazon EC2 또는 기타 사용자 지정 오리진 사용 63
오리진 그룹 사용 64
Amazon S3 콘텐츠가 있는 경우 CloudFront 추가 64
Amazon S3 버킷을 다른 리전으로 이동 66
사용자 지정 URL 사용 66
대체 도메인 이름 추가 67
대체 도메인 이름을 다른 CloudFront 배포로 이동 69
대체 도메인 이름 제거 72
CloudFront에 추가하는 대체 도메인 이름에서 와일드카드 사용 73
대체 도메인 이름 사용과 관련된 요구 사항 73
대체 도메인 이름에 대한 제한 74
Websockets 사용 75
WebSocket 프로토콜의 작동 방식 75
WebSocket 요구 사항 76
콘텐츠 추가, 제거 또는 바꾸기 77
콘텐츠 추가 및 액세스 77
기존 콘텐츠 업데이트 77
버전이 지정된 파일 이름을 사용하여 기존 파일 업데이트 78
동일한 파일 이름을 사용하여 기존 콘텐츠 업데이트 78
CloudFront가 콘텐츠를 배포하지 않도록 콘텐츠 제거 78
파일 URL 사용자 지정 79
고유의 도메인 이름 사용(Example.com) 79
URL에서 후행 슬래시(/) 사용 79
제한된 콘텐츠에 대한 서명된 URL 생성 80
RTMP 배포 미디어 파일에 대한 URL 생성 80
파일 무효화 80
파일을 무효화하는 방법과 버전 지정된 파일 이름을 사용하는 방법 중에서 선택 81
무효화할 파일 결정 81
무효화할 파일 지정 81
콘솔을 사용하여 파일 무효화 84
CloudFront API를 사용하여 파일 무효화 86
파일 무효화를 위한 타사 도구 86
동시 무효화 요청 최대값 86
파일 무효화에 대한 요금 결제 86
압축된 파일 제공 87
콘텐츠를 압축하도록 CloudFront 배포 구성 87
CloudFront를 사용하여 콘텐츠 압축 87
사용자 지정 오리진을 사용하여 콘텐츠 압축 90
Amazon S3 오리진을 사용하여 콘텐츠 압축 90
오리진 서버가 IIS를 실행 중일 때 압축된 파일 제공 91
오리진 서버가 NGINX를 실행 중일 때 압축된 파일 제공 91
보안 액세스 구성 및 콘텐츠에 대한 액세스 제한 92
CloudFront에서 HTTPS 사용 92
최종 사용자와 CloudFront 간에 HTTPS 필요 93
사용자 지정 오리진에 대해 HTTPS 요구 94
Amazon S3 오리진에 대해 HTTPS 요구 97
지원되는 프로토콜과 암호 98
HTTPS 연결 요금 102
대체 도메인 이름과 HTTPS 사용 102
CloudFront에서 HTTPS 요청을 제공하는 방식 선택 102
CloudFront에서 SSL/TLS 인증서를 사용하기 위한 요구 사항 104
CloudFront에서 SSL/TLS 인증서 사용 시의 할당량(최종 사용자와 CloudFront 간의 HTTPS만 해당) 107
대체 도메인 이름과 HTTPS 구성 108
SSL/TLS 인증서에서 퍼블릭 키 크기 확인 111

SSL/TLS 인증서에 대한 할당량 증가 112
SSL/TLS 인증서 회전 113
사용자 지정 SSL/TLS 인증서에서 기본 CloudFront 인증서로 되돌리기 114
사용자 지정 SSL/TLS 인증서를 전용 IP 주소 사용에서 SNI 사용으로 전환 114
서명된 URL과 서명된 쿠키를 사용하여 콘텐츠 제한 115
프라이빗 콘텐츠 제공 개요 115
작업 목록: 프라이빗 콘텐츠 제공 117
신뢰할 수 있는 서명자 지정 118
서명된 URL과 서명된 쿠키 중 선택 124
서명된 URL 사용 124
서명된 쿠키 사용 141
Base64 인코딩 및 암호화를 위한 Linux 명령 및 OpenSSL 사용 155
서명된 URL 코드 예제 156
Amazon S3 콘텐츠에 대한 액세스 제한 176
OAI 설정 개요 176
CloudFront OAI 생성 및 배포에 추가 177
OAI에 Amazon S3 버킷 내 파일 읽기 권한 부여 179
서명 버전 4 인증만 지원하는 Amazon S3 리전에서 OAI 사용 182
AWS AWS WAF를 사용하여 콘텐츠에 대한 액세스 제어 182
지리적 제한 콘텐츠 183
CloudFront 지리적 제한 사용 183
타사 지리적 위치 서비스 사용 184
필드 레벨 암호화 사용으로 민감한 데이터 보호 185
필드 레벨 암호화 개요 187
필드 레벨 암호화 설정 188
오리진의 데이터 필드 해독 191
콘텐츠 캐싱 및 가용성 최적화 193
캐싱과 엣지 캐시 193
캐시 적중률 개선 193
CloudFront에서 객체를 캐싱하는 시간 지정 194
쿼리 문자열 파라미터 기반 캐싱 194
쿠키 값 기반 캐싱 194
요청 헤더 기반 캐싱 195
압축이 불필요할 때 Accept-Encoding 헤더 제거 195
HTTP를 사용한 미디어 콘텐츠 제공 196
캐싱 및 쿼리 문자열 파라미터 196
쿼리 문자열 전달 및 캐싱에 대한 콘솔 및 API 설정 197
캐싱 최적화 197
쿼리 문자열 파라미터 및 CloudFront 액세스 로그 198
쿠키를 기반으로 콘텐츠 캐싱 198
요청 헤더에 따라 콘텐츠 캐싱 200
헤더 및 배포 - 개요 201
캐싱의 기반이 되는 헤더 선택 201
CORS 설정을 존중하도록 CloudFront 구성 202
디바이스 유형을 기반으로 캐싱하도록 구성 203
최종 사용자 언어를 기반으로 캐싱하도록 구성 203
최종 사용자 위치를 기반으로 캐싱하도록 구성 203
요청의 프로토콜을 기반으로 캐싱하도록 구성 203
압축 파일에 대한 캐싱 구성 203
헤더를 기반으로 한 캐싱이 성능에 미치는 영향 203
헤더 및 헤더 값의 대소문자가 캐싱이 미치는 영향 203
CloudFront에서 최종 사용자에게 반환하는 헤더 204
캐시 만료 관리 204
헤더를 사용하여 개별 객체의 캐시 기간 제어 205
CloudFront에서 웹 배포용 객체를 캐싱하는 시간 지정 206
CloudFront에서 RTMP 배포용 객체를 캐싱하는 최소 시간 지정 208
Amazon S3 콘솔을 사용하여 객체에 헤더 추가 209

오리진 장애 조치를 통한 고가용성 향상 ... 209
오리진 그룹 만들기 ... 212
Lambda@Edge 함수와 함께 오리진 장애 조치 사용 ... 212
사용자 지정 오류 페이지와 함께 오리진 장애 조치 사용 ... 214
범위 GET 처리 방법 ... 214
기본 루트 객체 지정 ... 215
기본 루트 객체를 지정하는 방법 ... 215
헤더가 기본 루트 객체에 대해 작동하는 방식 ... 216
루트 객체를 정의하지 않을 경우 CloudFront의 작동 방식 ... 217
문제 해결 ... 218
배포 문제 해결 ... 218
대체 도메인 이름을 추가하려고 하면 CloudFront에서 InvalidViewerCertificate 오류를 반환합니다 .. 218
배포에서 파일을 볼 수 없습니다. ... 219
오류 메시지: Certificate: <certificate-id> Is Being Used by CloudFront ... 220
오리진의 오류 응답 문제 해결 ... 221
HTTP 400 상태 코드(잘못된 요청) ... 221
HTTP 500 상태 코드(Lambda 실행 오류) ... 222
HTTP 502 상태 코드(잘못된 게이트웨이) ... 222
HTTP 502 상태 코드(Lambda 확인 오류) ... 224
HTTP 503 상태 코드(Lambda 제한 초과됨) ... 225
HTTP 503 상태 코드(Service Unavailable) ... 225
HTTP 504 상태 코드(게이트웨이 제한 시간) ... 226
CloudFront 로드 테스트 ... 228
요청 및 응답 동작 ... 230
Amazon S3 오리진에 대한 요청 및 응답 동작 ... 230
CloudFront에서 HTTP 및 HTTPS 요청을 처리하는 방법 ... 230
CloudFront에서 요청을 처리하고 Amazon S3 오리진 서버에 요청을 전달하는 방법 ... 230
CloudFront에서 Amazon S3 오리진 서버의 요청을 처리하는 방법 ... 234
사용자 지정 오리진에 대한 요청 및 응답 동작 ... 236
CloudFront에서 요청을 처리하고 사용자 지정 오리진 서버에 요청을 전달하는 방법 ... 236
CloudFront에서 사용자 지정 오리진 서버의 요청을 처리하는 방법 ... 244
오리진 그룹에 대한 요청 및 응답 동작 ... 247
사용자 지정 헤더를 오리진 요청에 추가 ... 248
오리진 사용자 지정 헤더의 사용 사례 ... 248
사용자 지정 헤더를 오리진 요청에 추가하도록 CloudFront 구성 ... 249
CloudFront가 오리진 요청에 추가할 수 없는 사용자 지정 헤더 ... 249
권한 부여 헤더를 전달하도록 CloudFront 구성 ... 250
CloudFront에서 오리진 서버의 HTTP 3xx 상태 코드를 처리하는 방법 ... 250
CloudFront에서 오리진의 HTTP 4xx 및 5xx 상태 코드를 처리 및 캐싱하는 방법 ... 250
사용자 지정 오류 페이지를 구성했을 때 CloudFront에서 오류를 처리하는 방법 ... 251
사용자 지정 오류 페이지를 구성하지 않았을 때 CloudFront에서 오류를 처리하는 방법 ... 252
CloudFront에서 캐싱하는 HTTP 4xx 및 5xx 상태 코드 ... 253
사용자 지정 오류 응답 생성 중 ... 255
특정 HTTP 상태 코드에 대한 사용자 지정 오류 페이지 만들기 ... 255
다른 위치에 저장 ... 257
응답 코드 변경 ... 257
오류 캐싱 방식 제어 ... 257
사용자 지정 오류 페이지를 사용할 수 없을 때 CloudFront에서 응답하는 방식 ... 258
사용자 지정 오류 페이지에 대한 요금 ... 259
오류 응답 동작 구성 ... 259
온디맨드 비디오 및 라이브 스트리밍 비디오 ... 261
스트리밍 비디오: 온디맨드 비디오 및 라이브 스트리밍 ... 261
온디맨드 비디오 제공 ... 262
Microsoft Smooth Streaming용 온디맨드 비디오 구성 ... 262
라이브 스트리밍 비디오 제공 ... 263
AWS Elemental MediaStore를 오리진으로 사용하여 비디오 제공 ... 264
AWS Elemental MediaPackage를 사용하여 포맷된 라이브 비디오 제공 ... 265

RTMP 배포 작업 268
RTMP 배포 269
RTMP 배포가 작동하는 방식 269
RTMP를 사용한 미디어 파일 스트리밍용 작업 목록 271
CloudFront 콘솔을 사용하여 RTMP 배포 만들기 271
RTMP 배포의 생성 또는 업데이트 시 지정하는 값 272
RTMP 배포의 생성 또는 업데이트 시 CloudFront에서 콘솔에 표시하는 값 276
미디어 플레이어 구성 277
RTMP 배포에 대해 Amazon S3 버킷을 오리진으로 사용 278
단일 오리진 서버에 대해 여러 RTMP 배포 생성 279
Crossdomain.xml을 사용한 액세스 제한 279
RTMP 배포 관련 오류 코드 279
RTMP 배포 문제 해결 280
Lambda@Edge를 사용하여 사용자 지정 281
Lambda@Edge 함수 생성 및 사용 시작하기 282
자습서: 간단한 함수 생성 283
IAM 권한 및 역할 설정 292
Lambda 함수를 CloudFront 배포와 연결하는 데 필요한 IAM 권한 293
서비스 보안 주체에 대한 함수 실행 역할 293
Lambda@Edge의 서비스 연결 역할 294
함수 작성 및 생성 298
Lambda@Edge의 함수 작성 298
Lambda 콘솔에서 Lambda@Edge 함수 생성 299
Lambda@Edge의 Lambda 함수 편집 300
프로그래밍 방식으로 Lambda 함수 및 CloudFront 트리거 생성 301
트리거 추가 302
Lambda 함수를 트리거할 수 있는 CloudFront 이벤트 302
Lambda 함수를 트리거하는 데 사용할 CloudFront 이벤트를 결정하는 방법 304
Lambda 콘솔을 사용하여 트리거 추가 304
CloudFront 콘솔을 사용하여 트리거 추가 305
테스트 및 디버깅 306
Lambda@Edge 함수 테스트 307
CloudFront에서 Lambda 함수 오류 파악 307
잘못된 Lambda 함수 응답(확인 오류) 문제 해결하기 312
Lambda 함수 실행 오류 문제 해결 313
Lambda@Edge 리전 확인 313
계정이 CloudWatch로 로그를 푸시하는지 확인 313
CloudWatch 지표 및 CloudWatch Logs 314
CloudWatch 지표 314
CloudWatch Logs 314
함수 및 복제본 삭제 315
이벤트 구조 315
동적 오리진 선택 315
이벤트 요청 316
응답 이벤트 321
요청 및 응답으로 작업 326
Lambda@Edge 함수와 오리진 장애 조치 사용 326
요청 트리거에서 HTTP 응답 생성 326
오리진 응답 트리거에서 HTTP 응답 업데이트 329
본문 포함 옵션을 선택해 요청 본문에 액세스 329
예제 함수 330
일반 예제 330
응답 생성 - 예제 333
쿼리 문자열 처리 - 예제 337
국가 또는 디바이스 유형 헤더에 따른 콘텐츠 개인 설정 - 예제 341
콘텐츠 기반 동적 오리진 선택 - 예제 345
오류 상태 업데이트 - 예제 352

요청 본문에 액세스 - 예 354
요구 사항 및 제한 357
CloudFront 배포 및 연결 358
Lambda 함수에 대한 CloudFront 트리거 358
CloudWatch Logs 358
헤더 358
HTTP Status Codes 360
Lambda 기능 지원 실행 시간 및 구성 360
할당량 361
Microsoft Smooth Streaming 362
네트워크 액세스 362
쿼리 문자열 매개 변수 362
본문 포함 옵션이 적용된 본문에 대한 최대 크기 363
태그 지정 363
URI 363
URI 및 쿼리 문자열 인코딩 363
보고서, 액세스 로그, 모니터링 365
CloudFront에 대한 AWS 결제 및 사용 보고서 365
CloudFront에 대한 AWS 결제 보고서 365
CloudFront에 대한 AWS 사용 보고서 366
CloudFront에 대한 AWS 청구서 및 AWS 사용 보고서 해석 367
CloudFront 콘솔 보고서 369
CloudFront 캐시 통계 보고서 371
CloudFront 인기 객체 보고서 374
CloudFront 상위 참조자 보고서 378
CloudFront 사용 보고서 380
CloudFront Viewers Reports 385
AWS Config를 통해 구성 변경 추적 392
CloudFront에서 AWS Config 설정 392
CloudFront 구성 기록 보기 393
액세스 로그 사용 393
로깅 작동 방식 394
액세스 로그용 Amazon S3 버킷 선택 394
로그 기록 및 로그 파일에 대한 액세스를 구성하는 데 필요한 권한 395
SSE-KMS 버킷에 사용할 경우 필요한 CMK 키 정책 396
파일 이름 형식 396
로그 파일 전송 시점 396
요청 URL 또는 헤더가 최대 크기를 초과했을 때 요청이 기록되는 방법 397
액세스 로그 분석 397
로깅 설정 편집 397
Amazon S3 버킷에서 로그 파일 삭제 398
로그 파일 형식 398
액세스 로그 요금 410
CloudTrail을 사용하여 API 요청 캡처 410
CloudTrail의 CloudFront 정보 411
CloudFront 로그 파일 항목 이해 412
Amazon CloudWatch를 사용하여 CloudFront 모니터링 416
CloudFront 및 Lambda@Edge 지표 보기 416
경보 설정 418
데이터 다운로드 419
프로그래밍 방식으로 지표 가져오기 421
Amazon CloudFront 배포에 태그 지정 424
태그 제한 424
배포에 태그 추가, 편집, 삭제 425
보안 426
데이터 보호 426
전송 중 데이터 암호화 427

- 저장 데이터 암호화 ... 427
- 콘텐츠에 대한 액세스 제한 ... 427
- Identity and Access Management ... 428
  - 인증 ... 428
  - 액세스 제어 ... 429
  - 액세스 관리 개요 ... 430
  - CloudFront에 대한 IAM 정책 사용 ... 434
  - CloudFront API 권한 참조 ... 439
- 로깅 및 모니터링 ... 444
- 규정 준수 확인 ... 444
  - CloudFront 규정 준수 모범 사례 ... 445
- 복원성 ... 446
  - CloudFront 오리진 장애 조치 ... 446
- 인프라 보안 ... 446

할당량 ... 447
- 일반 할당량 ... 447
- 웹 배포의 일반 할당량 ... 447
- WebSocket 연결에 대한 할당량 ... 448
- 화이트리스트에 등록된 쿠키에 대한 할당량(웹 배포만 해당) ... 448
- 화이트리스트에 등록된 쿼리 문자열에 대한 할당량(웹 배포만 해당) ... 449
- 사용자 지정 헤더에 대한 할당량(웹 배포만 해당) ... 449
- SSL 인증서에 대한 할당량(웹 배포만 해당) ... 449
- 무효화에 대한 할당량 ... 450
- 필드 레벨 암호화에 대한 할당량 ... 450
- Lambda@Edge에 대한 할당량 ... 450
  - URI 및 쿼리 문자열의 크기 할당량 ... 451
  - 본문 포함 옵션이 적용된 요청 본문에 대한 크기 할당량 ... 451
- 요청 제한 시간 ... 452
- RTMP 배포에 대한 할당량 ... 452

관련 정보 ... 453
- 추가 Amazon CloudFront 설명서 ... 453
- 지원 받기 ... 453
- CloudFront 개발자 도구 및 SDK ... 453
- Amazon Web Services 블로그에서 얻을 수 있는 팁 ... 454
- 객체 무효화 ... 454
- 비공개 콘텐츠 구성을 위한 도구 및 코드 예제 ... 454

문서 기록 ... 456

AWS Glossary ... 458

Amazon CloudFront란?

Amazon CloudFront는 .html, .css, .js 및 이미지 파일과 같은 정적 및 동적 웹 콘텐츠를 사용자에게 더 빨리 배포하도록 지원하는 웹 서비스입니다. CloudFront는 엣지 로케이션이라고 하는 데이터 센터의 전 세계 네트워크를 통해 콘텐츠를 제공합니다. CloudFront를 통해 서비스하는 콘텐츠를 사용자가 요청하면 지연 시간이 가장 낮은 엣지 로케이션으로 라우팅되므로 콘텐츠 전송 성능이 뛰어납니다.

- 콘텐츠가 이미 지연 시간이 가장 낮은 엣지에 있는 경우 CloudFront가 콘텐츠를 즉시 제공합니다.
- 콘텐츠가 엣지 로케이션에 없는 경우 CloudFront는 콘텐츠의 최종 버전에 대한 소스로 지정된 오리진—예: Amazon S3 버킷, MediaPackage 채널, HTTP 서버(예: 웹 서버) 등에서 콘텐츠를 검색합니다.

예를 들어, CloudFront가 아닌 일반적인 웹 서버에서 이미지를 제공한다고 가정합니다. 예를 들어 `http://example.com/sunsetphoto.png` URL을 사용하여 sunsetphoto.png라는 이미지를 서비스할 수 있습니다.

사용자는 이 URL로 쉽게 이동해 해당 이미지를 볼 수 있습니다. 하지만 이미지가 발견될 때까지 인터넷으로 이루어진 상호 연결된 네트워크의 복잡한 모음을 통해 네트워크에서 다른 네트워크로 요청이 라우팅되었다는 사실은 아마도 모르고 있을 것입니다.

CloudFront는 AWS 백본 네트워크를 통해 콘텐츠를 가장 효과적으로 서비스할 수 있는 엣지로 각 사용자 요청을 라우팅하여 콘텐츠 배포 속도를 높입니다. 일반적으로 CloudFront 엣지가 최종 사용자에게 가장 빨리 제공합니다. AWS 네트워크를 사용하면 사용자의 요청이 반드시 통과해야 하는 네트워크의 수가 줄어들어 성능이 향상됩니다. 파일의 첫 바이트를 로드하는 데 걸리는 지연 시간이 줄어들고 데이터 전송 속도가 빨라집니다.

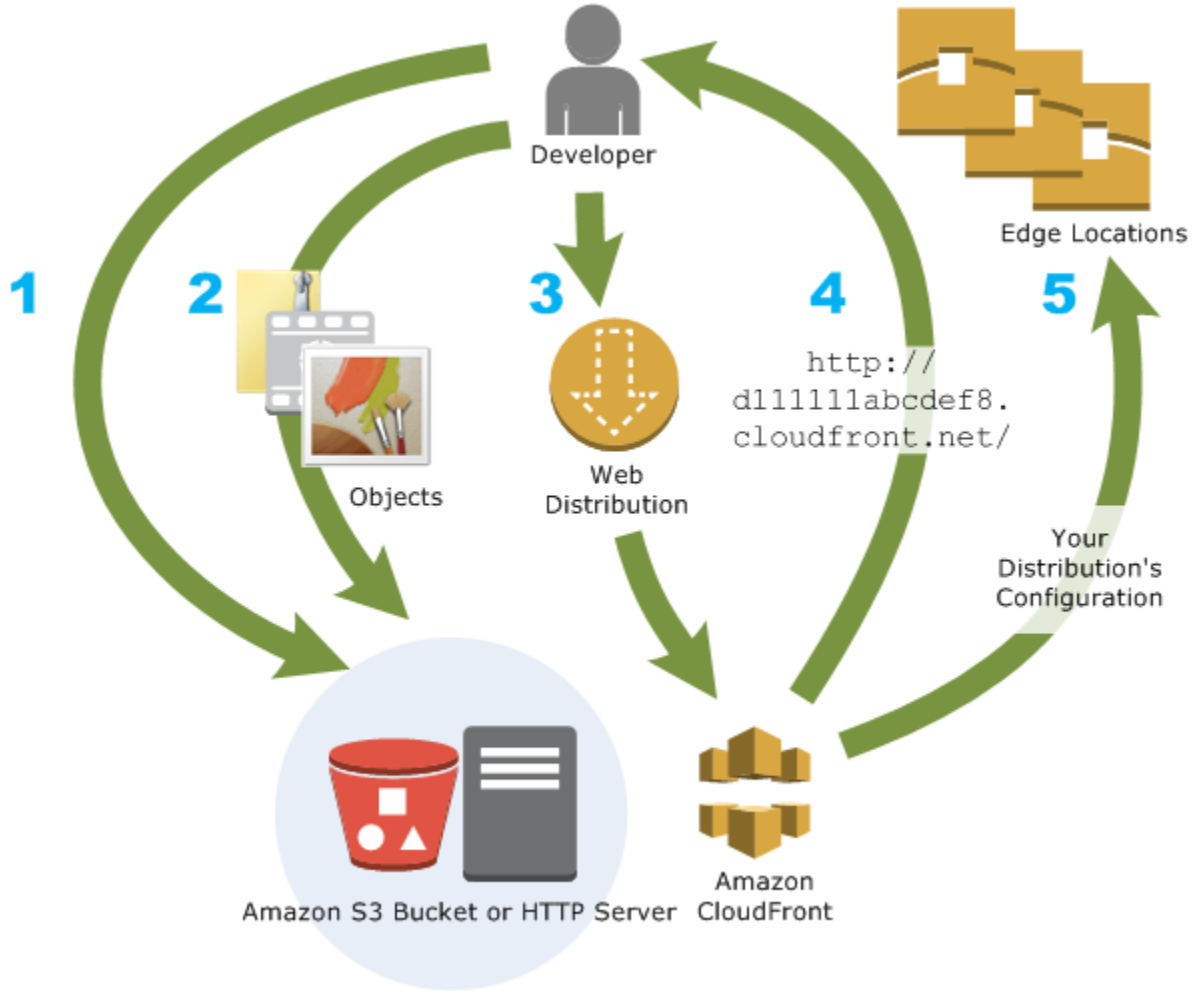
또한 파일(객체라고도 함)의 사본이 전 세계 여러 엣지 로케이션에 유지(또는 캐시)되므로 안정성과 가용성이 향상됩니다.

주제

- 콘텐츠를 전송하도록 CloudFront를 설정하는 방법 (p. 1)
- CloudFront 사용 사례 (p. 3)
- CloudFront에서 콘텐츠를 제공하는 방법 (p. 4)
- CloudFront 엣지 서버의 위치 및 IP 주소 범위 (p. 6)
- CloudFront에 액세스 (p. 7)
- 처음으로 Amazon CloudFront 사용하기 (p. 7)
- AWS Identity and Access Management (p. 7)
- CloudFront 요금 (p. 8)

콘텐츠를 전송하도록 CloudFront를 설정하는 방법

CloudFront 배포를 생성하여 CloudFront에게 어디로부터 콘텐츠를 전송하고자 하는지와 이러한 콘텐츠 전송을 추적 및 관리하는 방법에 대한 세부 정보를 알립니다. 그런 다음 CloudFront는 최종 사용자와 인접한 컴퓨터(엣지 서버)를 사용하여 사용자가 콘텐츠를 사용하거나 보고자 할 때 빠르게 전송합니다.



콘텐츠를 제공하도록 CloudFront를 구성하는 방법

1. Amazon S3 버킷 또는 고유 HTTP 서버와 같은 오리진 서버,를 지정하고, CloudFront는 이로부터 파일을 가져온 다음 전 세계 CloudFront 엣지 로케이션에 배포합니다.

   오리진 서버는 객체의 최종 원본 버전을 저장합니다. HTTP를 통해 콘텐츠를 서비스하는 경우 오리진 서버가 Amazon S3 버킷 또는 웹 서버 같은 HTTP 서버입니다. HTTP 서버는 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스나 사용자가 관리하는 서버에서 실행할 수 있습니다. 이 서버를 사용자 지정 오리진이라고도 합니다.

   Adobe Media Server RTMP 프로토콜을 사용해 미디어 파일을 주문형으로 배포할 경우 오리진 서버는 항상 Amazon S3 버킷입니다.

2. 오리진 서버에 파일을 업로드합니다. 객체라고도 하는 파일에는 일반적으로 웹 페이지, 이미지 및 미디어 파일이 포함되지만, HTTP 또는 Adobe Flash Media Server에서 사용되는 프로토콜에 해당하는 지원되는 Adobe RTMP 버전을 통해 서비스할 수 있다면 무엇이든지 포함될 수 있습니다.

   Amazon S3 버킷을 오리진 서버로 사용할 경우 버킷에 있는 객체를 공개적으로 읽을 수 있는 상태로 만들 수 있으므로 객체의 CloudFront URL을 아는 사람이라면 누구나 액세스할 수 있습니다. 객체를 비공개로 유지하고 액세스할 수 있는 사용자를 제어할 수 있는 옵션도 있습니다. 서명된 URL과 서명된 쿠키를 사용하여 프라이빗 콘텐츠 제공 (p. 115) 단원을 참조하십시오.

3. 사용자가 웹 사이트나 애플리케이션을 통해 파일을 요청할 경우 CloudFront에 어떤 오리진 서버에서 파일을 가져올지 알려 주는 CloudFront 배포를 만듭니다. 동시에 CloudFront에서 모든 요청을 기록할지, 배포를 만들자마자 활성화할지 여부와 같은 세부 사항을 지정합니다.
4. CloudFront는 새 배포에 도메인 이름을 할당하고, 이는 CloudFront 콘솔에서 볼 수 있습니다. 또는 API 요청 등과 같은 프로그램 요청에 대한 응답으로 반환됩니다. 원하는 경우 대신 사용할 대체 도메인 이름을 추가할 수 있습니다.
5. CloudFront에서는 배포의 구성(사용자의 콘텐츠가 아님)을 모든 해당 엣지 로케이션 또는 CloudFront가 파일의 사본을 캐시하는 지리적으로 분산된 데이터 센터의 POP(Point of Presence)— 서버 모음으로 보냅니다.

웹 사이트 또는 애플리케이션을 개발할 경우 CloudFront가 URL에 제공하는 도메인 이름을 사용합니다. 예를 들어 CloudFront가 `d111111abcdef8.cloudfront.net`를 배포의 도메인 이름으로 반환할 경우 Amazon S3 버킷(또는 HTTP 서버의 루트 디렉터리)에 있는 logo.jpg의 URL이 `http://d111111abcdef8.cloudfront.net/logo.jpg`입니다.

또는 CloudFront를 설정하여 고유한 도메인 이름을 배포와 사용할 수도 있습니다. 이 경우 URL이 `http://www.example.com/logo.jpg`가 될 수 있습니다.

또는 파일에 머리글을 추가하도록 오리진 서버를 구성할 수 있습니다. 이는 CloudFront 엣지 로케이션의 캐시에 파일을 얼마나 오래 보관할지 나타냅니다. 기본적으로 각 파일은 만료되기 전에 24시간 동안 엣지 로케이션에 남아 있습니다. 최소 만료 시간은 0초이며, 최대 만료 시간은 없습니다. 자세한 내용은 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204) 단원을 참조하십시오.

CloudFront 사용 사례

CloudFront를 사용하면 다양한 목표를 달성할 수 있습니다. 이 단원에서는 이에 대한 정보를 몇 가지 설명하고 추가 정보에 대한 링크를 제공합니다.

주제

- 정적 웹 사이트 콘텐츠 전송 속도 향상 (p. 3)
- 온디맨드 비디오 또는 라이브 스트리밍 비디오 제공 (p. 3)
- 시스템 처리를 통해 특정 필드 암호화 (p. 4)
- 엣지에서의 사용자 지정 (p. 4)
- Lambda@Edge 사용자 지정을 사용하여 프라이빗 콘텐츠 제공 (p. 4)

정적 웹 사이트 콘텐츠 전송 속도 향상

CloudFront는 전 세계 최종 사용자에게 제공되는 정적 콘텐츠(이미지, 스타일 시트, JavaScript 등)의 전송 속도를 높일 수 있습니다. CloudFront를 사용하면 AWS 백본 네트워크와 CloudFront 엣지 서버의 장점을 활용하여 해당 웹 사이트를 방문하는 최종 사용자에게 빠르고 안전하며 신뢰할 수 있는 환경을 제공할 수 있습니다.

정적 콘텐츠를 저장하고 전송하기 위한 간단한 방법은 Amazon S3 버킷을 사용하는 것입니다. S3와 CloudFront를 함께 사용하면 OAI(원본 액세스 ID)를 통해 S3 콘텐츠에 대한 액세스를 쉽게 제한할 수 있으며 이외에도 다양한 장점이 있습니다.

빠른 시작을 도와주는 AWS CloudFormation 템플릿을 비롯해 CloudFront와 S3를 함께 사용하는 방법에 대한 자세한 내용은 Amazon S3 + Amazon CloudFront: A Match Made in the Cloud 단원을 참조하십시오.

온디맨드 비디오 또는 라이브 스트리밍 비디오 제공

CloudFront는 전 세계 최종 사용자에게 미디어(녹화 파일 및 라이브 이벤트)를 스트리밍하기 위한 몇 가지 옵션을 제공합니다.

- 온디맨드 비디오(VOD) 스트리밍의 경우 CloudFront를 사용하면 MPEG DASH, Apple HLS, Microsoft Smooth Streaming, CMAF 등과 같은 일반적인 포맷으로 디바이스에 상관없이 스트리밍할 수 있습니다.
- 라이브 스트림 방송의 경우에는, 엣지에 미디어 조각을 캐싱하여 해당 조각을 올바른 순서로 전송하는 매니페스트 파일에 대한 여러 요청을 결합함으로써 오리진 서버의 부하를 줄일 수 있습니다.

CloudFront를 사용하여 스트리밍 콘텐츠를 전송하는 방법에 대한 자세한 내용은 CloudFront를 사용한 온디맨드 비디오 및 라이브 스트리밍 비디오 (p. 261) 단원을 참조하십시오.

시스템 처리를 통해 특정 필드 암호화

CloudFront에서 HTTPS를 구성할 경우 오리진 서버에 대한 종단 간 연결의 보안이 보장됩니다. 파일 수준 암호화를 추가하는 경우, HTTPS 보안뿐 아니라 시스템 처리 전체에서 특정 데이터를 보호함으로써 오리진의 특정 애플리케이션만 데이터를 보게 할 수 있습니다.

필드 레벨 암호화를 설정하려면 CloudFront에 퍼블릭 키를 추가한 후 이 키를 사용하여 암호화하려는 필드 세트를 지정합니다. 자세한 내용은 필드 레벨 암호화 사용으로 민감한 데이터 보호 (p. 185) 단원을 참조하십시오.

엣지에서의 사용자 지정

엣지에서 서버리스 코드를 실행하면 지연 시간을 최소화하면서 다양한 방식으로 최종 사용자에 대한 콘텐츠와 환경을 사용자 지정할 수 있습니다. 예를 들어 오리진 서버가 유지보수를 위해 다운된 경우 최종 사용자에게 일반 HTTP 오류 메시지를 표시하는 대신 사용자 지정 오류 메시지를 표시할 수 있습니다. 또는 함수를 사용하여 CloudFront가 오리진으로 요청을 전송하기 전에 사용자를 인증하고 콘텐츠에 대한 액세스를 제어할 수 있습니다.

Lambda@Edge를 CloudFront와 함께 사용하면 CloudFront가 전송하는 콘텐츠를 다양한 방법으로 사용자 지정할 수 있습니다. Lambda@Edge에 대한 자세한 내용과 CloudFront를 사용하여 함수를 만들고 배포하는 방법을 알아보려면 Lambda@Edge를 사용하여 엣지에서 콘텐츠 사용자 지정 (p. 281) 단원을 참조하십시오. 해당 솔루션에 맞게 사용자 지정할 수 있는 다양한 코드 샘플을 보려면 Lambda@Edge 함수 예제 (p. 330) 단원을 참조하십시오.

Lambda@Edge 사용자 지정을 사용하여 프라이빗 콘텐츠 제공

Lambda@Edge를 사용하면 서명된 URL 및 쿠키 사용 옵션으로, 자체 사용자 지정 오리진에서 프라이빗 콘텐츠를 제공하도록 CloudFront 배포를 구성할 수 있습니다.

오리진에 대한 액세스를 CloudFront 전용으로 제한하는 방법에는 몇 가지가 있으며, 방화벽에 CloudFront IP 화이트리스트 사용하거나, 공유 암호를 전달하는 사용자 지정 헤더를 사용하는 방법 등이 있습니다.

샘플 코드를 비롯한 단계별 지침에 대한 자세한 내용은 Amazon CloudFront 및 AWS Lambda@Edge를 사용한 프라이빗 콘텐츠 제공 단원을 참조하십시오.

CloudFront에서 콘텐츠를 제공하는 방법

초기 설정 후에 CloudFront는 해당 웹 사이트나 애플리케이션과 상호 작용하여 콘텐츠 전송 속도를 높입니다. 이 단원에서는 최종 사용자가 콘텐츠를 요청할 때 CloudFront가 콘텐츠를 어떻게 제공하는지를 설명합니다.

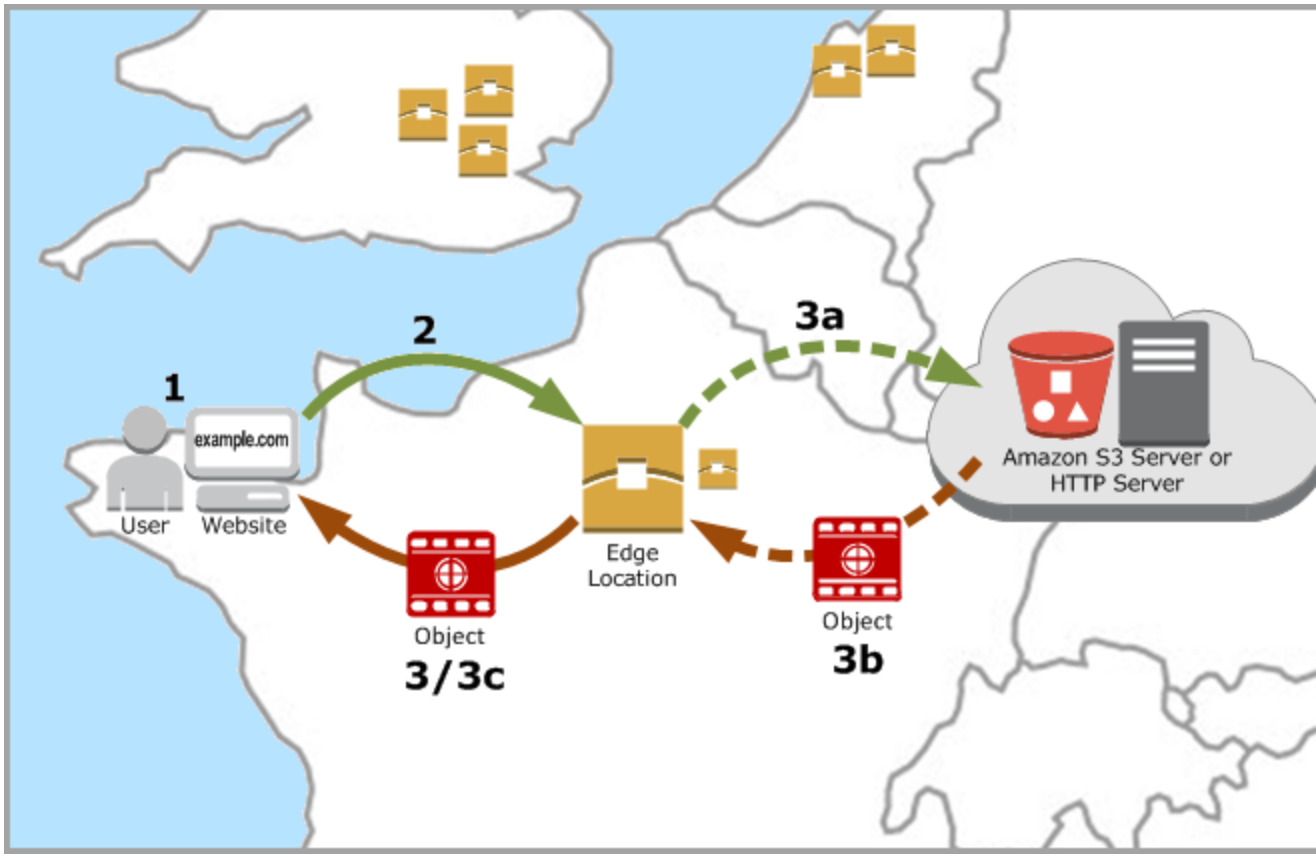
주제

- CloudFront에서 사용자에게 콘텐츠를 제공하는 방법 (p. 5)
- CloudFront에서 리전 엣지 캐시를 사용하는 방식 (p. 5)

CloudFront에서 사용자에게 콘텐츠를 제공하는 방법

콘텐츠를 제공하도록 CloudFront를 구성한 후에 사용자가 파일을 요청하면 다음과 같은 결과를 얻습니다.

1. 사용자가 웹 사이트 또는 애플리케이션에 액세스하고 이미지 파일 및 HTML 파일 같은 하나 이상의 파일을 요청합니다.
2. DNS가 요청을 최적으로 서비스할 수 있는 CloudFront POP(엣지 로케이션)으로 요청을 라우팅합니다.—이 위치는 일반적으로 지연 시간과 관련해 가장 가까운 CloudFront POP이며—요청을 해당 엣지 로케이션으로 라우팅합니다.
3. POP에서 CloudFront는 해당 캐시에 요청된 파일이 있는지 확인합니다. 파일이 캐시에 있으면 CloudFront는 파일을 사용자에게 반환합니다. 파일이 캐시에 없으면 다음을 수행합니다.
   a. CloudFront는 배포의 사양과 요청을 비교하고 파일에 대한 요청을 해당하는 파일 형식으로 사용자의 오리진 서버(—예: 이미지 파일의 경우 Amazon S3 버킷 및 HTML 파일의 경우 HTTP 서버)에 전달합니다.
   b. 오리진 서버는 파일을 다시 엣지 로케이션으로 보냅니다.
   c. 오리진에서 첫 번째 바이트가 도착하면 CloudFront가 파일을 사용자에게 전달하기 시작합니다. CloudFront는 다음에 다른 사용자가 해당 파일을 요청할 때 엣지 로케이션의 캐시에 파일을 추가합니다.



CloudFront에서 리전 엣지 캐시를 사용하는 방식

CloudFront 입지 지점(POP)(엣지 로케이션)은 많이 사용되는 콘텐츠를 최종 사용자에게 빠르게 서비스할 수 있도록 합니다. CloudFront에는 또한 리전 엣지 캐시가 있어 POP에 있을 정도로 많이 사용되는 콘텐츠가 아

닌 경우에도 최종 사용자가 보다 많은 콘텐츠를 가까이에서 액세스할 수 있도록 하여 콘텐츠의 성능을 개선하는 데 도움을 줍니다.

리전 엣지 캐시는 모든 유형의 콘텐츠, 특히 시간이 지나면서 점차 사용되지 않게 되는 콘텐츠에 유용합니다. 이러한 콘텐츠의 예로는 동영상, 사진 또는 아트워크와 같은 사용자 생성 콘텐츠, 제품 사진 및 동영상과 같은 전자 상거래 자산, 갑자기 사용자가 많아질 수 있는 뉴스 및 이벤트 관련 콘텐츠 등이 있습니다.

리전 캐시 작동 방식

리전 엣지 캐시는 최종 사용자가 쉽게 접할 수 있도록 전역으로 배포되는 CloudFront 위치입니다. 최종 사용자에게 콘텐츠를 직접적으로 제공하는 POP—전역적 엣지 로케이션과 오리진 서버 사이에 위치합니다. 객체의 사용도가 떨어지면 개별 POP가 이러한 객체를 제거하여 보다 많이 사용되는 콘텐츠가 해당 공간을 사용하도록 합니다. 리전 엣지 캐시는 개별 POP보다 캐시가 더 큽니다. 따라서 객체가 가장 가까운 리전 엣지 캐시 위치에 더 오래 캐시 상태로 유지됩니다. 이에 따라 최종 사용자에게 보다 가까운 위치에 더 많은 콘텐츠를 유지할 수 있으며, CloudFront가 오리진 서버로 되돌아갈 필요가 없고 최종 사용자의 전반적인 능률이 향상됩니다.

최종 사용자가 웹 사이트나 애플리케이션을 통해 요청하면, DNS는 이 요청을 사용자의 요청을 가장 잘 처리할 수 있는 POP로 라우팅합니다. 일반적으로 지연 시간을 기준으로 가장 가까운 CloudFront 엣지 로케이션이 여기에 해당합니다. POP에서 CloudFront는 해당 캐시에 요청된 파일이 있는지 확인합니다. 파일이 캐시에 있으면 CloudFront는 파일을 사용자에게 반환합니다. 파일이 캐시에 없으면, POP가 가장 가까운 리전 엣지 캐시로 이동하여 객체를 가져옵니다.

리전 엣지 캐시 위치에서 CloudFront는 해당 캐시에 요청된 파일이 있는지 다시 확인합니다. 파일이 캐시에 있으면 CloudFront가 파일을 요청한 POP에 전달합니다. 리전 엣지 캐시 위치에서 첫 번째 바이트가 도착하면 CloudFront가 파일을 사용자에게 전달하기 시작합니다. CloudFront는 다음에 다른 사용자가 해당 파일을 요청할 때 POP의 캐시에 파일을 추가합니다.

파일이 POP 또는 리전 엣지 캐시 위치에서 캐시되지 않는 경우, CloudFront는 배포의 사양과 요청을 비교하고 파일에 대한 요청을 오리진 서버로 전달합니다. 오리진 서버가 파일을 리전 엣지 캐시 위치로 다시 보낸 이후, 해당 파일이 POP에 전달되고, CloudFront가 이를 사용자에게 전달합니다. 이 경우 CloudFront는 다음 번에 최종 사용자가 해당 파일을 요청할 때 POP 뿐만 아니라 리전 엣지 캐시 위치의 캐시에도 파일을 추가합니다. 이를 통해 리전의 모든 POP에서 로컬 캐시를 공유하도록 하고, 오리진 서버에 요청이 여러 번 제출되지 않습니다. CloudFront는 또한 오리진 서버와의 연결을 계속 유지하므로 최대한 빨리 오리진으로부터 해당 파일을 가져올 수 있습니다.

Note

- 리전 엣지 캐시는 POP에 기능 패리티를 둡니다. 예를 들어, 캐시 무효화 요청이 들어오면 객체가 만료되기 전에 POP 캐시 및 리전 엣지 캐시 모두에서 객체가 제거됩니다. 다음에 최종 사용자가 객체를 요청할 때, CloudFront에서는 최신 버전의 객체를 가져오도록 오리진에 반환합니다.
- 프록시 메서드 `PUT/POST/PATCH/OPTIONS/DELETE`는 POP에서 오리진으로 바로 이동하며 리전 엣지 캐시를 통해 프록시되지 않습니다.
- 요청 시간에 결정되는 동적 콘텐츠(모든 헤더를 전달하도록 구성된 캐시 동작)는 리전 엣지 캐시를 통해 이동하지 않으며, 오리진으로 바로 이동합니다.

CloudFront 엣지 서버의 위치 및 IP 주소 범위

CloudFront 엣지 서버의 위치 목록은 Amazon CloudFront 제품 세부 정보 페이지를 참조하십시오.

Amazon Web Services(AWS)는 현재 IP 주소 범위를 JSON 형식으로 게시합니다. 현재 범위를 보려면 ip-ranges.json을 다운로드합니다. 자세한 내용은 Amazon Web Services 일반 참조의 AWS IP 주소 범위를 참조하십시오.

CloudFront 엣지 서버와 연결된 IP 주소 범위를 찾으려면 ip-ranges.json에서 다음 문자열을 검색합니다.

```
"service": "CLOUDFRONT"
```

또는 여기에서 CloudFront IP 범위만 볼 수 있습니다.

CloudFront에 액세스

Amazon CloudFront에 액세스하는 방법은 다음과 같습니다.

- AWS Management 콘솔 – 이 가이드의 절차는 AWS Management 콘솔을 사용하여 작업을 수행하는 방법을 설명합니다.
- AWS SDK –AWS가 SDK를 제공하는 프로그래밍 언어를 사용하는 경우, SDK를 사용하여 CloudFront에 액세스할 수 있습니다. SDK는 인증을 단순화하고, 개발 환경에 쉽게 통합되며, CloudFront 명령에 액세스할 수 있도록 합니다. 자세한 내용은 Amazon Web Services용 도구를 참조하십시오.
- CloudFront API – SDK가 제공되지 않는 프로그래밍 언어를 사용하는 경우, Amazon CloudFront API Reference에서 API 작업에 대한 정보와 API 요청을 수행하는 방법을 참조하십시오.
- AWS Command Line Interface – 자세한 내용은 AWS Command Line Interface 사용 설명서의 AWS Command Line Interface 설정하기를 참조하십시오.
- Windows PowerShell용 AWS 도구 – 자세한 내용은 Windows PowerShell용 AWS 도구 사용 설명서의 Windows PowerShell용 AWS 도구 설정하기를 참조하십시오.

처음으로 Amazon CloudFront 사용하기

Amazon CloudFront 시작하기에 대한 자세한 내용은 이 가이드의 다음 단원을 참조하십시오.

- Amazon CloudFront 설정 (p. 10) - AWS 가입 방법, AWS 계정에 대한 보안 액세스 방법, CloudFront에 대한 프로그래밍 방식 액세스 구성 방법을 설명합니다.
- Amazon CloudFront 시작하기 (p. 14) - Amazon S3 버킷이나 웹 사이트 등과 같은 오리진에서 최종 사용자에게 콘텐츠를 제공할 수 있는 배포를 만드는 방법과 배포를 테스트하는 방법을 설명합니다.

AWS Identity and Access Management

Amazon CloudFront은 다음과 같은 작업을 수행할 수 있는 서비스인 AWS Identity and Access Management(IAM)와 통합됩니다.

- 조직의 AWS 계정에서 사용자와 그룹을 생성합니다.
- 계정 사용자 간에 간편하게 AWS 계정 리소스 공유
- 각 사용자에게 고유한 보안 자격 증명을 할당합니다.
- 서비스 및 리소스에 대한 사용자 액세스 상세 제어

예를 들어, CloudFront를 IAM과 함께 사용하여 AWS 계정에서 새로운 배포를 만들거나 캐시 동작 설정을 업데이트할 수 있는 사용자를 제어할 수 있습니다.

IAM에 대한 전반적인 정보는 다음을 참조하십시오.

- CloudFront의 Identity and Access Management (p. 428)
- Identity and Access Management(IAM)

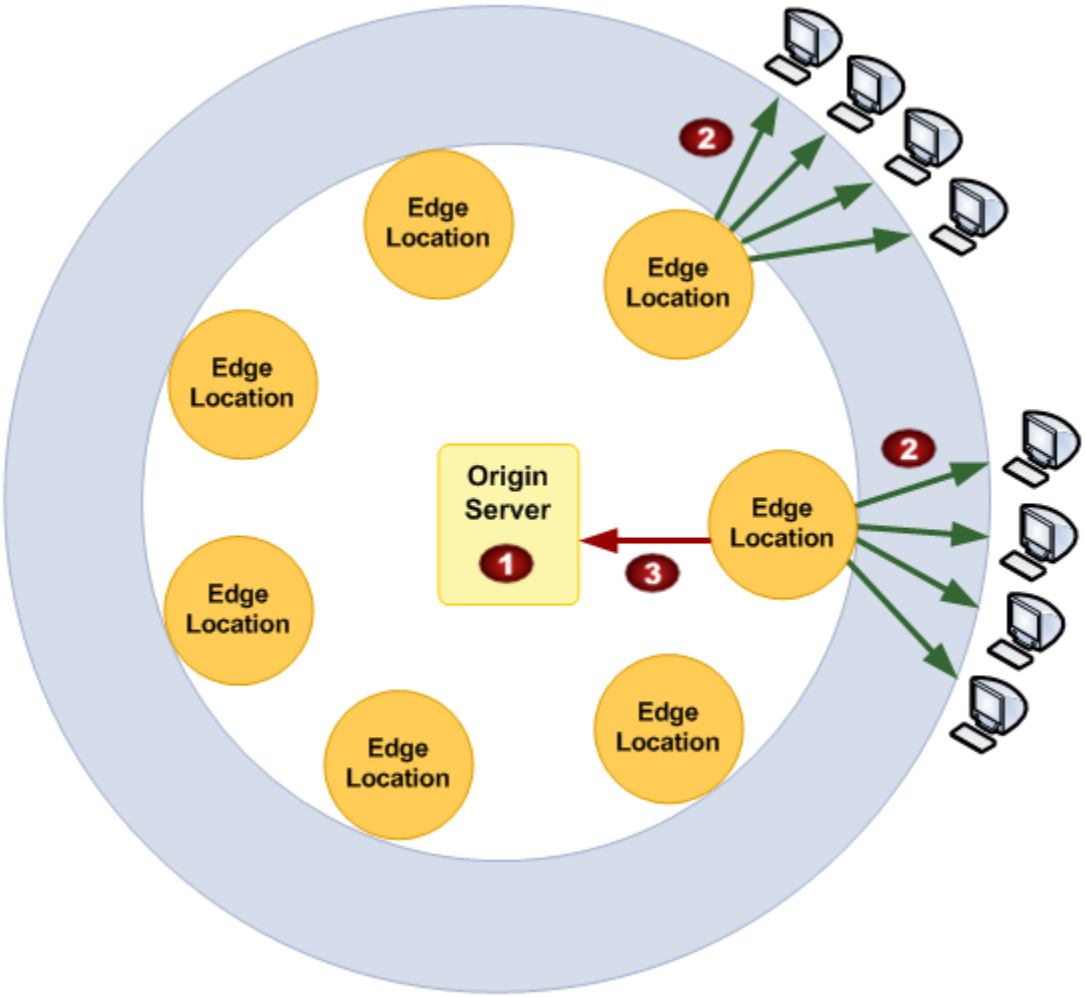
- IAM 사용 설명서

CloudFront 요금

Amazon CloudFront는 사용자가 선불 요금을 지불하거나 얼마나 많은 콘텐츠를 보유할지 약정하지 않아도 되도록 디자인되었습니다. 기타 AWS 서비스와 마찬가지로 사용할 때, 사용한 내역에 대해서만 요금을 지불합니다.

AWS는 CloudFront에 대해 두 가지 사용 보고서를 제공합니다. 결제 보고서와 사용 활동을 요약하는 보고서입니다. 이러한 보고서에 대한 자세한 내용은 CloudFront에 대한 AWS 결제 및 사용 보고서 (p. 365) 단원을 참조하십시오.

다음 다이어그램과 목록에는 CloudFront를 사용하는 데 드는 요금이 요약되어 있습니다.



AWS에서 보내는 월 청구서에는 AWS 서비스와 기능에 따라 사용량 및 달러 액수가 구분되어 있습니다. 다음에는 이전 그림에 설명된 요금이 나열되어 있습니다. 자세한 내용은 Amazon CloudFront 요금을 참조하십시오.

1. Amazon S3 버킷에서의 스토리지에 대한 요금. 버킷에 객체를 보관하는 데 일반 Amazon S3 스토리지 요금을 지불합니다. 요금은 AWS 청구서의 Amazon S3 부분에 나타납니다.

2. 엣지 로케이션에서 객체 서비스에 대한 요금. CloudFront가 객체 요청에 응답할 때 CloudFront 요금이 발생합니다. 요금에는 WebSocket 데이터의 서버에서 클라이언트로의 전송에 따른 금액이 포함됩니다. CloudFront 요금은 AWS 청구서의 CloudFront 부분에 ## -DataTransfer-Out-Bytes로 표시됩니다.
3. 오리진에 데이터 제출에 대한 요금 청구. 사용자가 `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청을 포함하는 데이터를 오리진에 전송할 경우 CloudFront 요금이 발생합니다. 요금에는 WebSocket 데이터의 클라이언트에서 서버로의 전송에 따른 금액이 포함됩니다. CloudFront 요금은 AWS 청구서의 CloudFront 부분에 ## -DataTransfer-Out-OBytes로 표시됩니다.

다음에 유의하십시오.

- HTTPS 요청에 대한 추가 요금도 발생하며, 이 요청에 대한 추가 요금 또한 필드 레벨 암호화가 활성화되어 있습니다. 자세한 내용은 Amazon CloudFront 요금을 참조하십시오.
- 오리진 그룹을 사용할 때에는 추가적인 CloudFront 요금이 발생하지 않습니다. CloudFront를 다른 AWS 또는 비 AWS 오리진과 함께 사용할 때와 동일한 요청 수수료 및 데이터 전송 요금을 계속 지불하게 됩니다. 자세한 내용은 CloudFront 오리진 그룹 사용 (p. 64) 단원을 참조하십시오.

CloudFront 배포의 가격 등급 선택

CloudFront는 전 세계에 엣지 로케이션를 보유하고 있습니다. 각 엣지 로케이션에 대한 비용이 각기 다르므로 개발자에게 부과되는 가격도 CloudFront에서 요청을 처리하는 엣지 로케이션에 따라 다릅니다.

CloudFront 엣지 로케이션은 지리적 리전으로 그룹화되며 리전은 가격 등급으로 그룹화됩니다. 기본 가격 등급에는 모든 리전이 포함됩니다. 또 다른 가격 등급에는 대부분 리전(미국, 캐나다, 유럽, 홍콩, 필리핀, 한국, 대만, 싱가포르, 일본, 인도, 남아프리카 및 중동 리전)이 포함되지만 최고가 리전은 제외됩니다. 세 번째 가격 등급에는 최저가 리전(미국, 캐나다, 유럽 리전)만 포함됩니다.

기본적으로 CloudFront는 성능에만 기초하여 객체에 대한 요청에 응답합니다. 즉, 해당 최종 사용자에 대한 지연 시간이 가장 짧은 엣지 로케이션으로부터 객체가 제공됩니다. 일부 지리적 리전에 있는 최종 사용자에 대해 지연 시간이 좀더 길더라도 보다 저렴한 비용으로 제공하려면 일부 CloudFront 리전만을 포함하는 가격 등급을 선택할 수 있습니다. CloudFront가 해당 가격 등급에 속하는 엣지 로케이션에서만 객체를 제공하지만, 여전히 선택한 가격 등급 내 엣지 로케이션에서 지연 시간이 가장 낮은 엣지 로케이션에서 콘텐츠를 제공합니다. 하지만 일부 최종 사용자(특히 가격 등급 내 속하지 않는 지리적 리전에 있는 최종 사용자)의 경우 모든 CloudFront 엣지 로케이션에서 콘텐츠가 제공될 때보다는 지연 시간이 더 길 수 있습니다. 예를 들어 미국 및 유럽만을 포함하는 가격 등급을 선택하는 경우 호주와 아시아의 최종 사용자는 호주와 아시아를 포함하는 가격 등급을 선택할 때보다 더 긴 지연 시간을 경험해야 할 수도 있습니다.

선택한 가격 등급에 일부 엣지 로케이션이 포함되지 않는 경우 CloudFront에서 이따금씩 가격 등급에 포함되지 않은 리전 내 엣지 로케이션에 있는 콘텐츠에 대한 요청을 처리해야 할 수도 있습니다. 이러한 상황이 발생할 경우 객체가 제공되는 리전의 요율이 더 높더라도 이 요율로 요금이 부과되지는 않습니다. 대신에 선택한 가격 등급에서 최저가 리전의 요율로 부과됩니다.

CloudFront 배포를 생성하거나 업데이트할 때 가격 등급을 선택할 수 있습니다. 자세한 내용은 배포 작업 (p. 31) 단원을 참조하십시오.

CloudFront API, AWS SDK 중 하나, 또는 AWS CloudFormation을(를) 사용하여 배포를 생성하거나 업데이트하는 경우 DistributionConfig 복합 형식 단원을 참조하십시오(`PriceClass`를 검색).

CloudFront 요금 및 가격 등급에 대한 자세한 내용은 Amazon CloudFront 요금 단원을 참조하십시오.

Amazon CloudFront 설정

이 섹션의 개요와 절차는 AWS를 시작하는 데 도움이 됩니다.

주제

- AWS에 가입 (p. 10)
- 계정에 액세스 (p. 10)
- IAM 사용자 생성 (p. 11)
- AWS Command Line Interface 또는 Windows PowerShell용 AWS 도구 설정 (p. 13)
- AWS SDK 다운로드 (p. 13)

AWS에 가입

AWS에 가입하면 Amazon CloudFront를 포함한 AWS의 모든 서비스에 AWS 계정이 자동으로 등록됩니다. 사용한 서비스에 대해서만 청구됩니다.

이미 AWS 계정이 있으면 계정에 액세스 (p. 10) 단계로 건너뜁니다. 계정이 없으면 만드십시오.

AWS 계정을 생성하려면

1. https://portal.aws.amazon.com/billing/signup을 엽니다.
2. 온라인 지시 사항을 따릅니다.

   등록 절차 중 전화를 받고 전화 키패드를 사용하여 확인 코드를 입력하는 과정이 있습니다.

나중에 필요하므로 AWS 계정 번호를 기록합니다.

> **Tip**
>
> CloudFront를 사용하여 S3 버킷에 저장한 콘텐츠를 배포할 계획이라면 S3 가입 단계도 꼭 완료해야 합니다. 자세한 내용은 Amazon S3에 가입을 참조하십시오.

계정에 액세스

다음 옵션을 사용하여 AWS 서비스를 이용합니다.

- AWS Management 콘솔
- 각 서비스의 API
- AWS Command Line Interface (AWS CLI)
- Windows PowerShell용 AWS 도구
- AWS SDK

각 옵션의 경우, 서비스 사용 권한이 있음을 확인하는 자격 증명을 입력하여 AWS 계정에 액세스해야 합니다.

콘솔에 액세스

AWS 관리 콘솔에 처음 액세스하려면 이메일 주소와 암호를 입력합니다. 이러한 이메일 주소와 암호의 조합을 루트 자격 증명 또는 루트 계정 자격 증명이라고 합니다. 계정에 처음 액세스한 후에는 일상적 사용에 루트 계정 자격 증명을 다시 사용하지 말 것을 강력히 권장합니다. 그 대신 AWS Identity and Access Management를 사용하여 새 자격 증명을 만들어야 합니다. 그러려면 IAM 사용자라는 본인의 사용자 계정을 만든 다음 IAM 사용자를 관리자 권한이 있는 IAM 그룹에 추가하거나 IAM 사용자에게 관리자 권한을 부여하십시오. 그러면 IAM 사용자의 특정 URL이나 자격 증명을 사용하여 AWS에 액세스할 수 있습니다. 나중에 다른 IAM 사용자를 추가하고 이들의 액세스를 계정의 지정된 리소스로 제한할 수도 있습니다.

> **Note**
>
> 웹 브라우저에 대한 일부 광고 차단 플러그인은 Amazon CloudFront 콘솔 작업을 방해하므로 이로 인해 콘솔이 예상치 못한 작동을 보일 수 있습니다. 웹 브라우저에 광고 차단 플러그인을 설치한 경우에는, 플러그인의 화이트리스트에 CloudFront 콘솔의 URL인 https://console.aws.amazon.com/cloudfront/home을 추가하는 것이 좋습니다.

API, AWS CLI, Windows PowerShell용 AWS 도구 또는 AWS SDK에 액세스

API, AWS CLI, Windows PowerShell용 AWS 도구 또는 AWS SDK에 액세스 하려면 액세스 키를 만들어야 합니다. 이들 키는 액세스 키 ID 및 보안 액세스 키로 이루어져 있는데, 이를 사용하여 AWS에 보내는 프로그래밍 방식의 요청에 서명할 수 있습니다.

키를 만들려면 AWS Management 콘솔에 로그인합니다. 루트 자격 증명 대신 IAM 사용자 자격 증명을 사용하여 로그인하는 것이 좋습니다. 자세한 내용은 IAM 사용 설명서에서 IAM 사용자의 액세스 키 관리를 참조하십시오.

IAM 사용자 생성

다음 절차를 수행하여 관리자 그룹을 만들고 IAM 사용자를 만든 다음 IAM 사용자를 관리자 그룹에 추가합니다. AWS에 등록하였지만 자신의 IAM 사용자를 아직 생성하지 않았다면 IAM 콘솔에서 생성할 수 있습니다. 콘솔을 사용하는 데 익숙하지 않은 경우, AWS Management Console로 작업하기의 개요를 참조하십시오.

관리자 사용자를 직접 생성하여 관리자 그룹에 추가하려면(콘솔)

1. https://console.aws.amazon.com/iam/에서 AWS 계정 이메일 주소 및 비밀번호를 AWS 계정 루트 사용자로 사용하여 IAM 콘솔에 로그인합니다.

   > **Note**
   >
   > **`Administrator`** IAM 사용자를 사용하는 아래 모범 사례를 준수하고, 루트 사용자 자격 증명을 안전하게 보관해 두는 것이 좋습니다. 몇 가지 계정 및 서비스 관리 작업을 수행하려면 반드시 루트 사용자로 로그인해야 합니다.

2. 탐색 창에서 사용자와 사용자 추가를 차례로 선택합니다.
3. 사용자 이름에 **`Administrator`**를 입력합니다.
4. AWS Management 콘솔 액세스 옆의 확인란을 선택합니다. 그런 다음 Custom password(사용자 지정 암호)를 선택하고 텍스트 상자에 새 암호를 입력합니다.
5. (선택 사항) 기본적으로 AWS에서는 새 사용자가 처음 로그인할 때 새 암호를 생성해야 합니다. User must create a new password at next sign-in(사용자가 다음에 로그인할 때 새 암호를 생성해야 합니다) 옆에 있는 확인란의 선택을 취소하면 새 사용자가 로그인한 후 암호를 재설정할 수 있습니다.
6. Next: Permissions(다음: 권한)를 선택합니다.

7. 권한 설정 아래에서 그룹에 사용자 추가를 선택합니다.
8. 그룹 생성을 선택합니다.
9. 그룹 생성 대화 상자의 그룹 이름에 **`Administrators`**를 입력합니다.
10. Filter policies(정책 필터링)을 선택한 다음 AWS managed -job function(AWS 관리형 -job 함수)을 선택하여 테이블 내용을 필터링합니다.
11. 정책 목록에서 AdministratorAccess 확인란을 선택합니다. 그런 다음 Create group을 선택합니다.

    Note

    `AdministratorAccess` 권한을 사용하여 AWS Billing and Cost Management 콘솔에 액세스하려면 먼저 결제에 대한 IAM 사용자 및 역할 액세스를 활성화해야 합니다. 이를 위해 결제 콘솔에 액세스를 위임하기 위한 자습서 1단계의 지침을 따르십시오.
12. 그룹 목록으로 돌아가 새 그룹의 확인란을 선택합니다. 목록에서 그룹을 확인하기 위해 필요한 경우 Refresh(새로 고침)를 선택합니다.
13. Next: Tags(다음: 태그)를 선택합니다.
14. (선택 사항) 태그를 키-값 페어로 연결하여 메타데이터를 사용자에게 추가합니다. IAM에서 태그 사용에 대한 자세한 내용은 IAM 사용 설명서의 IAM 개체 태그 지정을 참조하십시오.
15. Next: Review(다음: 검토)를 선택하여 새 사용자에 추가될 그룹 멤버십의 목록을 확인합니다. 계속 진행할 준비가 되었으면 Create user를 선택합니다.

이와 동일한 절차에 따라 그룹 및 사용자를 추가로 생성하고 사용자에게 AWS 계정 리소스에 액세스할 수 있는 권한을 부여할 수 있습니다. 특정 AWS 리소스에 대한 사용자 권한을 제한하는 정책을 사용하는 방법을 알아보려면 액세스 관리 및 정책 예제를 참조하십시오.

새 IAM 사용자로 로그인하려면

1. AWS 콘솔에서 로그아웃합니다.
2. 다음 URL을 사용하여 로그인합니다. 여기서 *`your_aws_account_id`*는 하이픈이 없는 본인의 AWS 계정 번호입니다. 예를 들어 사용자의 AWS 계정 번호가 1234-5678-9012인 경우, AWS 계정 ID는 123456789012입니다.

   ```
   https://your_aws_account_id.signin.aws.amazon.com/console/
   ```
3. 방금 생성한 IAM 사용자 이름(이메일 주소가 아님)과 암호를 입력합니다. 로그인하면 탐색 모음에 "*`your_user_name`* @ *`your_aws_account_id`*"가 표시됩니다.

로그인 페이지의 URL에 AWS 계정 ID가 포함되지 않게 하려면 계정 별칭을 생성합니다.

계정 별칭을 만들고 계정 ID를 숨기려면

1. IAM 콘솔의 탐색 창에서 대시보드를 선택합니다.
2. 대시보드에서 사용자 지정을 선택하고 회사 이름 등의 별칭을 입력합니다.
3. AWS 콘솔에서 로그아웃합니다.
4. 다음 URL을 사용하여 로그인합니다.

   ```
   https://your_account_alias.signin.aws.amazon.com/console/
   ```

본인 계정의 IAM 사용자 로그인 링크를 확인하려면 IAM 콘솔을 열고 대시보드에서 IAM 사용자 로그인 링크 아래를 확인합니다.

IAM 사용에 관한 자세한 내용은 CloudFront의 Identity and Access Management (p. 428) 단원을 참조하십시오.

AWS Command Line Interface 또는 Windows PowerShell용 AWS 도구 설정

AWS Command Line Interface(AWS CLI)은 AWS 서비스를 관리하는 통합 도구입니다. AWS CLI의 설치 및 구성 방법에 대한 자세한 내용은 AWS Command Line Interface 사용 설명서의 AWS Command Line Interface 초기 설정을 참조하십시오.

Windows PowerShell을 사용한 경험이 있다면 Windows PowerShell용 AWS 도구을 사용하는 것이 좋습니다. 자세한 내용은 Windows PowerShell용 AWS 도구 사용 설명서의 Windows PowerShell용 AWS 도구 설정하기를 참조하십시오.

AWS SDK 다운로드

AWS에서 SDK를 제공하는 프로그래밍 언어를 사용하는 경우, Amazon CloudFront API 대신 SDK를 쓰는 것이 좋습니다. SDK를 사용하면 인증을 더 간단하게 만들고, 개발 환경에 쉽게 통합할 수 있으며, CloudFront 명령에 쉽게 액세스할 수 있습니다. 자세한 내용은 AWS 기반의 도구를 참조하십시오.

Amazon CloudFront 시작하기

간단한 CloudFront 배포를 만들거나 AWS for WordPress 플러그인을 사용하여 CloudFront로 콘텐츠를 제공하는 기본적인 단계를 시작합니다. 이미 WordPress 웹 사이트가 있으면 AWS for WordPress 플러그인을 사용하여 CloudFront 배포를 만드는 것이 좋습니다. WordPress를 사용하지 않는다면 간단한 CloudFront 배포로 시작해 보십시오.

다음 두 옵션 모두 AWS 프리 티어 사용을 무료로 시작할 수 있습니다. 12개월 무료 사용 기간이 만료되거나 사용량이 프리 티어 범위를 초과하면 사용한 서비스에 대해 표준 종량 요금이 청구됩니다.

주제

- 간단한 CloudFront 배포로 시작하기 (p. 14)
- AWS for WordPress 플러그인 시작하기 (p. 22)

간단한 CloudFront 배포로 시작하기

이 단원에 나온 절차에서는 CloudFront를 사용하여 다음을 수행하는 기본 구성을 설정하는 방법을 보여줍니다.

- 하나의 Amazon Simple Storage Service(Amazon S3) 버킷에 객체의 원래 버전을 저장합니다.
- 텍스트 또는 그래픽 같은 콘텐츠를 배포합니다.
- 모든 사용자가 객체에 액세스할 수 있도록 합니다.
- 객체에 대한 URL에 CloudFront 도메인 이름을 사용합니다(예: `http://d111111abcdef8.cloudfront.net/image.jpg`).
- 기본 24시간(최소 시간 0초) 동안 CloudFront 엣지에 객체를 보관합니다.

이러한 옵션은 대부분 사용자 지정이 가능합니다. 예를 들어 S3 버킷을 사용하는 대신 웹 서버에 콘텐츠를 저장할 수 있으며, 서명된 URL이나 쿠키를 사용하여 콘텐츠를 액세스할 수 있는 사용자를 제한할 수 있습니다. CloudFront 배포 옵션을 사용자 지정하는 방법은 배포 만들기 단계(개요) (p. 34) 단원을 참조하십시오.

몇 가지 기본 단계만 수행하면 CloudFront를 사용하여 콘텐츠를 제공할 수 있습니다. 첫 번째 단계는 가입입니다. 그 다음에는 CloudFront 배포를 만들고 웹 페이지나 애플리케이션의 URL에 CloudFront 도메인 이름을 사용하여 콘텐츠를 참조합니다.

주제

- 사전 조건 (p. 14)
- 1단계: Amazon S3에 콘텐츠 업로드 및 객체 권한 부여 (p. 14)
- 2단계: CloudFront 배포 만들기 (p. 16)
- 3단계: 링크 테스트 (p. 21)

사전 조건

시작하기 전에 먼저 Amazon CloudFront 설정 (p. 10)의 단계를 완료해야 합니다.

1단계: Amazon S3에 콘텐츠 업로드 및 객체 권한 부여

Amazon S3 버킷은 파일(객체) 또는 폴더를 포함할 수 있는 컨테이너입니다. CloudFront는 Amazon S3 버킷을 원본으로 사용하여 사용자를 위해 거의 모든 파일 유형을 배포할 수 있습니다. 예를 들어 CloudFront에서

는 텍스트, 이미지 및 비디오를 배포할 수 있습니다. 여러 개의 버킷을 만들 수 있으며, Amazon S3에 저장할 수 있는 데이터의 양에는 최대값이 없습니다.

기본적으로 Amazon S3 버킷과 그 안에 있는 모든 파일은 비공개 상태로, 버킷을 만든 AWS 계정만 파일을 읽거나 쓸 수 있는 권한이 있습니다. 누구나 CloudFront URL을 사용해 Amazon S3 버킷에 있는 파일에 액세스할 수 있도록 허용하려면 객체에 공개 읽기 권한을 부여해야 합니다. 이는 CloudFront 및 Amazon S3으로 작업할 때 가장 일반적으로 범하는 실수 중 하나입니다. Amazon S3 버킷의 각 객체에 명시적으로 권한을 부여해야 합니다.

> **Note**
>
> 콘텐츠를 다운로드할 수 있는 사용자를 제한하기 위해 CloudFront 비공개 콘텐츠 기능을 사용할 수 있습니다. 비공개 콘텐츠를 배포하는 방법에 대한 자세한 내용은 서명된 URL과 서명된 쿠키를 사용하여 프라이빗 콘텐츠 제공 (p. 115)를 참조하십시오.

Amazon S3에 콘텐츠를 업로드하고 모두에게 읽기 권한을 부여하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/s3/에서 Amazon S3 콘솔을 엽니다.
2. Amazon S3 콘솔에서 버킷 만들기를 선택합니다.
3. 버킷 만들기 대화 상자의 이름 및 리전 페이지에서 다음을 수행합니다.
   a. 버킷 이름을 입력합니다.

      > **Important**
      >
      > 버킷을 CloudFront와 함께 사용하려면 DNS 명명 요구 사항을 준수하는 이름을 사용해야 합니다. 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 버킷 규제 및 제한을 참조하십시오.

   b. 리전에서 버킷의 AWS 리전을 선택합니다. 기본적으로 Amazon S3은 미국 동부(버지니아 북부) 리전에 버킷을 만듭니다. 지연 시간을 최적화하거나 비용을 최소화하려면 가까운 리전을 선택하는 것이 좋습니다. 또는 규제 요구 사항을 해소하기 위해 다른 리전을 선택할 수도 있습니다.
4. 다음을 선택합니다.
5. 옵션 구성 페이지에서 버전 관리, 태그 지정 및 기타 기능의 옵션을 선택합니다.
6. 다음을 선택합니다.
7. 권한 설정 페이지에서 다음 확인란을 지웁니다.

   - 모든 퍼블릭 액세스 차단

   CloudFront URL이 버킷의 콘텐츠를 제공할 수 있도록 버킷 및 파일에 대한 퍼블릭 읽기 액세스를 허용해야 합니다. 그러나 CloudFront 프라이빗 콘텐츠 기능을 사용하여 특정 콘텐츠에 대한 액세스를 제한할 수 있습니다. 자세한 내용은 서명된 URL과 서명된 쿠키를 사용하여 프라이빗 콘텐츠 제공 (p. 115) 단원을 참조하십시오.
8. [Next]를 선택한 다음 [Create bucket]을 선택합니다.
9. 버킷 창에서 버킷을 선택하고 업로드를 선택합니다.
10. 파일 선택 페이지에서 파일을 버킷에 끌어다 놓습니다. 또는 파일 추가를 선택한 후 업로드할 파일을 선택합니다.
11. 다음을 선택합니다.
12. 권한 설정 페이지에서 Amazon S3 버킷에 업로드하는 각 파일에 대한 퍼블릭 읽기 권한을 부여합니다.
    a. 다음을 선택하여 권한을 설정합니다.
    b. 퍼블릭 권한 관리에서 이 객체에 퍼블릭 읽기 액세스 권한을 부여함을 선택합니다.
    c. 다음을 선택합니다.
13. 암호화 또는 태깅과 같이 객체에 원하는 속성을 설정하고 다음을 선택합니다.
14. Upload를 선택합니다.

업로드가 완료된 후에 해당 URL을 사용하여 항목으로 이동할 수 있습니다. 예:

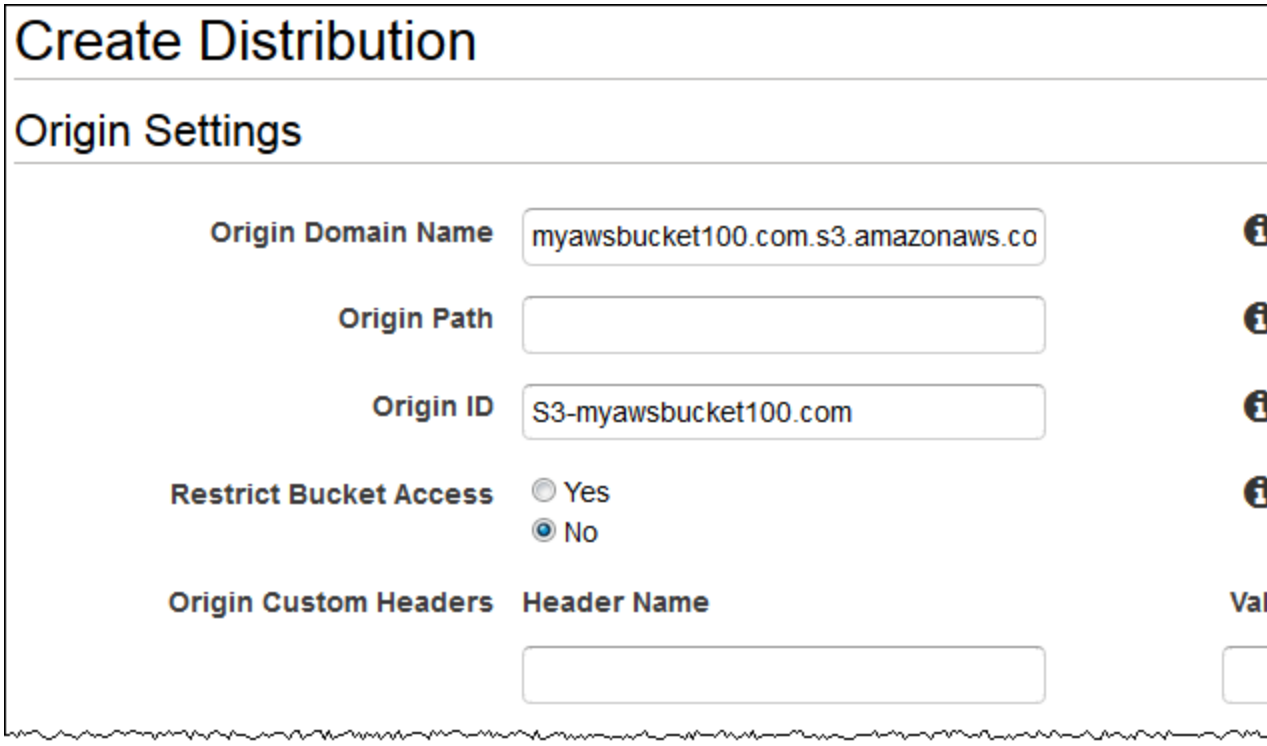
```
https://<bucket name>.s3-<AWS Region>.amazonaws.com/<object name>
```

Amazon S3 URL을 사용해 콘텐츠를 공개적으로 액세스할 수 있는지 확인하되, CloudFront에서 콘텐츠를 배포할 준비가 되었을 때 사용하는 URL은 이 URL이 아닙니다.

2단계: CloudFront 배포 만들기

CloudFront 배포를 만들려면

1. https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. [Create Distribution]을 선택합니다.
3. [Select a delivery method for your content]의 [Web] 섹션에서 [Get Started]를 선택합니다.
4. Create Distribution(배포 생성) 페이지의 Origin Domain Name(오리진 도메인 이름)에서 전에 만든 Amazon S3 버킷을 선택합니다. Origin Path(오리진 경로), Origin ID(오리진 ID), Restrict Bucket Access(버킷 액세스 제한) 및 Origin Custom Headers(오리진 사용자 지정 헤더)의 기본값을 그대로 사용합니다.



5. Default Cache Behavior Settings(기본 캐시 동작 설정)에서 기본값을 그대로 사용합니다.

   CloudFront는 다음을 수행합니다.

   - 배포를 위해 CloudFront URL(예: `http://d111111abcdef8.cloudfront.net/image.jpg`)을 사용하는 모든 요청을 이 절차의 4단계에서 지정한 Amazon S3 버킷에 전달합니다.
   - 최종 사용자가 HTTP 또는 HTTPS를 사용해 객체에 액세스할 수 있도록 허용합니다.
   - 객체에 대한 요청에 응답합니다.
   - 24시간 동안 CloudFront 엣지에 객체를 캐시합니다.
   - 기본 요청 헤더만 오리진으로 전달하고 헤더의 값에 따라 객체를 캐시하지는 마십시오.

- 객체에 대한 요청을 오리진에 전달할 때 쿠키와 쿼리 문자열 파라미터가 있는 경우 이를 제외합니다. Amazon S3은 쿼리를 처리하지 않으며 일부 쿼리 문자열 파라미터만 처리합니다.
- 모두가 콘텐츠를 볼 수 있도록 허용합니다.

CloudFront는 다음을 수행하지 않습니다.

- Microsoft Smooth Streaming 형식으로 미디어 파일을 배포하도록 구성됩니다.
- 콘텐츠를 자동으로 압축합니다.

캐시 동작 옵션에 대한 자세한 내용은 캐시 동작 설정 (p. 43)을 참조하십시오.

Default Cache Behavior Settings

Path Pattern	Default (*)
Viewer Protocol Policy	(●) HTTP and HTTPS ( ) Redirect HTTP to HTTPS ( ) HTTPS Only
Allowed HTTP Methods	(●) GET, HEAD ( ) GET, HEAD, OPTIONS ( ) GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE
Cached HTTP Methods	GET, HEAD (Cached by default)
Forward Headers	None (Improves Caching)
Object Caching	(●) Use Origin Cache Headers ( ) Customize Learn More
Minimum TTL	0
Maximum TTL	31536000
Default TTL	86400
Forward Cookies	None (Improves Caching)
Forward Query Strings	( ) Yes (●) No (Improves Caching)
Smooth Streaming	( ) Yes (●) No
Restrict Viewer Access (Use Signed URLs or Signed Cookies)	( ) Yes (●) No
Compress Objects Automatically	( ) Yes (●) No Learn More

6. Distribution Settings(배포 설정)에서 배포에 대한 값을 선택합니다.

   요금 계층

   CloudFront 서비스에 지불할 최고 요금에 해당하는 요금 등급을 선택합니다. 기본적으로 CloudFront는 모든 CloudFront 리전의 엣지 로케이션에서 객체를 제공합니다.

   요금 등급과 요금 등급 선택이 배포의 CloudFront 성능에 미치는 영향에 대한 자세한 내용은 CloudFront 배포의 가격 등급 선택 (p. 9) 단원을 참조하십시오. 요금 등급이 CloudFront 리전에 매핑되는 방식을 비롯해 CloudFront 요금에 대한 자세한 내용은 Amazon CloudFront 요금을 참조하십시오.

   AWS WAF 웹 ACL

   기본값 [None]을 수락합니다.

   Alternate Domain Names (CNAMEs)

   기본값을 수락합니다. 즉, 필드를 비워둡니다.

   SSL 인증서

   기본값인 기본 CloudFront 인증서(*.cloudfront.net)를 수락합니다.

   Supported HTTP Versions

   기본값인 HTTP/2, HTTP/1.1, HTTP/1.0을 수락합니다.

   기본 루트 객체(선택 사항)

   최종 사용자가 배포의 객체(`https://d111111abcdef8.cloudfront.net/product-description.html`)가 아닌 배포의 루트 URL(`https://d111111abcdef8.cloudfront.net`)을 요청할 때 CloudFront에서 오리진(예: `index.html`)으로부터 요청할 객체입니다. 기본 루트 객체를 지정하면 배포 콘텐츠가 노출되지 않습니다.

   로깅(선택 사항)

   기본값 Off를 수락합니다.

   IPv6 사용

   기본값을 수락합니다.

   설명(선택 사항)

   배포와 함께 저장할 설명을 입력합니다.

   Distribution State

   활성을 선택합니다.

Distribution Settings

Price Class Use All Edge Locations (Best Performance)

AWS WAF Web ACL None

Alternate Domain Names (CNAMEs)

SSL Certificate

- Default CloudFront Certificate (*.cloudfront.net)

  Choose this option if you want your users to use HTTPS or HTTP
  with the CloudFront domain name (such as https://d111111abcde
  /logo.jpg).
  Important: If you choose this option, CloudFront requires that bro
  support TLSv1 or later to access your content.

- Custom SSL Certificate (stored in AWS IAM): No certif

  Choose this option if you want your users to use HTTPS to acces
  alternate domain name (such as https://www.example.com/logo.
  dedicated CloudFront IP addresses or SNI.
  To choose this option, you first need to upload your certificate to t
  store (the -path parameter must start with /cloudfront/).
  Learn More

Default Root Object

Logging
- On
- Off

Bucket for Logs

Log Prefix

Cookie Logging
- On
- Off

Comment

Distribution State
- Enabled
- Disabled

Cancel Back Cre

7. [Create Distribution]을 선택합니다.
8. CloudFront에서 배포를 생성하면 배포의 상태 열 값이 진행 중에서 배포 완료로 변경됩니다. 처리에는 보통 20 ~ 40분이 소요됩니다.

   CloudFront가 배포에 할당하는 도메인 이름이 배포 목록에 나타납니다. 선택한 배포에 대한 [General] 탭에도 나타납니다.

3단계: 링크 테스트

배포를 생성하면 CloudFront가 Amazon S3 오리진 서버의 위치를 인식하므로 배포와 연결된 도메인 이름을 알 수 있습니다. 해당 도메인 이름을 사용해 Amazon S3 버킷 콘텐츠에 대한 링크를 만들고 CloudFront에서 이 링크를 서비스하도록 만들 수 있습니다.

> **Note**
>
> 링크를 테스트하기 전에 배포의 상태가 배포 완료로 변경될 때까지 기다려야 합니다.

객체에 연결하는 방법

1. 다음 HTML을 새 파일에 복사합니다.
   - <도메인 이름>을 CloudFront에서 배포에 할당한 도메인 이름으로 바꿉니다.
   - <객체 이름>을 Amazon S3 버킷의 파일 이름으로 바꿉니다.

```
<html>
<head>My CloudFront Test</head>
<body>
<p>My text content goes here.</p>
<p><img src="https://<domain name>/<object name>" alt="my test image"/>
</body>
</html>
```

   예를 들어 도메인 이름이 `d111111abcdef8.cloudfront.net`이고 객체가 `image.jpg`인 경우 링크의 URL은 다음과 같습니다.

   `https://d111111abcdef8.cloudfront.net/image.jpg`를 선택하십시오.

   객체가 버킷 안에 있는 폴더에 있는 경우 URL에 폴더를 포함하세요. 예를 들어 `image.jpg`가 이미지 폴더에 있으면 URL은 다음과 같습니다.

   `https://d111111abcdef8.cloudfront.net/images/image.jpg`
2. 파일 이름 확장명이 .html인 파일에 텍스트를 저장합니다.
3. 브라우저에서 웹 페이지를 열어 콘텐츠를 볼 수 있는지 확인합니다. 콘텐츠가 보이지 않으면 모든 단계를 제대로 수행했는지 확인합니다. 문제 해결 (p. 218)에서 팁을 확인할 수도 있습니다.

브라우저가 엣지에서 제공한 이미지 파일이 포함된 페이지를 반환합니다. 이 엣지는 CloudFront에서 객체를 서비스하기에 적합한 것으로 확인된 위치입니다.

CloudFront 사용에 관한 자세한 내용은 Amazon CloudFront 관련 정보 (p. 453) 단원을 참조하십시오.

AWS for WordPress 플러그인 시작하기

AWS for WordPress 플러그인을 사용하여 Amazon CloudFront를 비롯한 여러 AWS 서비스를 설정할 수 있습니다. CloudFront를 사용하면 WordPress 웹 사이트 방문자가 전 세계 엣지 로케이션에 캐시된 콘텐츠를 가속화된 환경에서 볼 수 있습니다. 방문자가 웹 사이트에 오면 CloudFront가 지연 시간을 최소화하여 보다 빠르고 안정된 환경을 즐길 수 있는 엣지 로케이션으로 방문자를 라우팅합니다.

AWS for WordPress 플러그인은 여러 캐시 동작을 이용해 웹 사이트에 있는 다양한 유형의 콘텐츠를 처리하여 WordPress 웹 사이트에 최적화된 CloudFront 배포를 만듭니다. 플러그인의 CloudFront 기능은 WordPress.com에서 호스팅되는 웹 사이트 및 Amazon Lightsail, Amazon EC2 또는 다른 웹 호스팅 플랫폼에서 자체 호스팅되는 WordPress 웹 사이트와 연동됩니다.

이 플러그인을 사용하여 Amazon Polly 및 Amazon Translate와 같은 다른 AWS 서비스를 설정한 후, 이 서비스에서 생성된 콘텐츠를 가속화하도록 CloudFront를 구성할 수도 있습니다. 플러그인을 이용한 Amazon Polly 설정에 대한 자세한 내용은 Amazon Polly 개발자 안내서의 Amazon Polly용 WordPress 플러그인을 참조하십시오. CloudFront를 사용하여 Amazon Polly에서 생성된 콘텐츠를 가속화하는 방법에 대한 자세한 내용은 (선택 사항) Amazon Polly 콘텐츠용 CloudFront 배포 만들기 (p. 28) 단원을 참조하십시오.

주제

- 사전 조건 (p. 22)
- 1단계: 플러그인 설치 (p. 24)
- 2단계: 플러그인으로 CloudFront 구성 및 사용 (p. 25)
- (선택 사항) 사이트 가속화 비활성화 (p. 27)
- (선택 사항) 사이트 가속화 제거 및 CloudFront 배포 삭제 (p. 27)
- (선택 사항) 플러그인 비활성화 및 제거 (p. 28)
- (선택 사항) Amazon Polly 콘텐츠용 CloudFront 배포 만들기 (p. 28)
- 문제 해결 (p. 29)

사전 조건

AWS for WordPress 플러그인을 사용하려면 AWS 계정, AWS Identity and Access Management(IAM) 사용자 및 WordPress 웹 사이트가 필요합니다.

주제

- AWS 계정 생성 (p. 22)
- IAM 사용자 생성 (p. 23)
- WordPress 웹 사이트 만들기 (p. 24)

AWS 계정 생성

AWS 계정이 이미 있으면 이 단원을 건너뛰어도 됩니다. 계정이 없으면 만드십시오.

AWS 계정을 만들려면 다음을 수행합니다.

1. https://portal.aws.amazon.com/billing/signup을 엽니다.
2. 온라인 지시 사항을 따릅니다.

   등록 절차 중 전화를 받고 전화 키패드를 사용하여 확인 코드를 입력하는 과정이 있습니다.

IAM 사용자 생성

AWS for WordPress 플러그인을 사용하려면 플러그인의 IAM 사용자를 만들어야 합니다. IAM 사용자는 AWS 서비스에 대한 API 호출을 수행할 권한이 있는 AWS 계정에 속한 사람 또는 애플리케이션입니다.

> **Note**
>
> WordPress.com을 사용하지 않고 Amazon EC2에 자체 호스팅되는 WordPress 웹 사이트가 있으면 IAM 사용자 대신 IAM 역할을 사용해도 됩니다. 자세한 내용은 Amazon EC2 사용 설명서의 Amazon EC2에 대한 IAM 역할 을 참조하십시오.

다음 절차에는 IAM 정책을 IAM 사용자에게 연결하는 단계가 포함되어 있습니다. IAM 정책은 사용자에 적용되는 권한을 정의하는 문서입니다.

IAM 사용자를 생성하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/iam/에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 사용자를 선택합니다. 그런 다음 Add user(사용자 추가)를 선택합니다.
3. Set user details(사용자 세부 정보 설정) 페이지에서 다음을 수행합니다.
   a. 사용자 이름에 **`AWSForWordPressPlugin`**을 입력합니다.
   b. Access type(액세스 유형)에서 Programmatic access(프로그래밍 방식 액세스)를 선택합니다.
   c. Next: Permissions(다음: 권한)를 선택합니다.
4. 권한 설정 페이지에서 다음을 수행합니다.
   a. Attach existing policies directly(기존 정책 직접 연결)를 선택합니다.
   b. 검색 상자에 **`WordPress`**를 입력한 다음 AWSForWordPressPluginPolicy 옆에 있는 확인란을 선택합니다.

      > **Note**
      >
      > AWSForWordPressPluginPolicy는 AWS 관리형 정책이며 사용자에게 AWS for WordPress 플러그인에 있는 기능을 모두 사용할 수 있는 권한을 부여합니다. 플러그인에 새 기능이 추가되면 AWS가 새 기능을 사용하는 데 필요한 권한을 포함하도록 이 정책을 업데이트합니다.

   c. 다음: 태그를 선택합니다.
5. [Next: Review]를 선택합니다.
6. Create user를 선택합니다.
7. Download .csv(.csv 다운로드)를 선택하여 사용자의 자격 증명(액세스 키 ID 및 보안 액세스 키)을 컴퓨터에 저장합니다. AWS for WordPress 플러그인을 .구성할 때 필요합니다.

   > **Important**
   >
   > 여기서만 사용자의 보안 액세스 키를 저장할 수 있으므로 지금 저장해야 합니다.

IAM 사용자 자격 증명 보호

앞 단원에서 만든 IAM 사용자는 AWS 계정에서 다음을 수행할 수 있습니다.

- `"createdBy" : "AWSForWordPressPlugin"` 태그를 사용하여 CloudFront 배포를 생성, 수정, 태그 지정, 나열 및 삭제하고 이 배포의 무효화를 만들고 나열합니다.
- 미국 동부(버지니아 북부) 지역에서 AWS Certificate Manager 인증서를 요청, 태그 지정, 나열 및 삭제합니다.
- 미국 동부(버지니아 북부) 지역에 AWS CloudFormation 스택을 만들고 `"createdBy" : "AWSForWordPressPlugin"` 태그를 사용하여 스택을 수정, 나열 및 삭제합니다.

- Amazon Polly를 사용하여 텍스트를 음성으로 변환하고 사용 가능한 Amazon Polly 음성을 나열합니다.
- Amazon Translate를 사용하여 텍스트를 다른 언어로 번역합니다.
- 특정 Amazon S3 버킷이 존재하는지 확인합니다.
- 이름이 `audio_for_wordpress` 또는 `audio-for-wordpress`로 시작하는 Amazon S3 버킷을 만들고 이 버킷에서 객체를 생성, 삭제 및 나열합니다.

> **Important**
>
> 권한이 없는 사용자가 이 권한을 얻을 수 없게 하려면 IAM 사용자의 자격 증명을 보호하십시오. 보안 액세스 키를 암호처럼 취급하고 안전한 장소에 보관하되, 다른 사람과 공유하지 마십시오. 암호와 마찬가지로 액세스 키를 주기적으로 바꾸십시오. 실수로 보안 액세스 키가 유출되면 즉시 삭제합니다. 그런 다음 새 액세스 키를 만들어 AWS for WordPress 플러그인에 사용할 수 있습니다.

WordPress 웹 사이트 만들기

이미 WordPress 웹 사이트가 있으면 1단계: 플러그인 설치 (p. 24) 단원으로 건너뛰어도 좋습니다.

WordPress 웹 사이트가 없으면 WordPress.com을 사용하여 만들 수 있습니다. AWS for WordPress 플러그인을 사용하려면 WordPress.com Business 또는 eCommerce 플랜이 필요합니다.

Amazon Lightsail, Amazon EC2 또는 다른 웹 호스팅 플랫폼을 사용하여 자체 웹 서버에 WordPress 소프트웨어를 설치할 수도 있습니다. 자체 WordPress 웹 사이트를 호스팅하면 WordPress.com을 사용할 때보다 많은 단계가 포함되며 웹 서버, 로드 밸런서, DNS 레코드 및 웹 서버 인증서를 구성하고 관리할 수 있어야 합니다.

WordPress 웹 사이트 설정 방법에 관계없이 AWS for WordPress 플러그인을 사용하려면 다음 사항이 필요합니다.

- 웹 사이트에 고유한 도메인 이름이 있어야 합니다. 웹 주소나 URL(uniform resource locator)이라고도 하는 도메인 이름은 방문자가 웹 사이트에 갈 때 사용하는 주소입니다. 예를 들어 Amazon의 도메인 이름은 amazon.com입니다. 이 주제에서는 example.com을 도메인 이름의 일반적인 예로 사용하지만 특정 웹 사이트에는 그에 맞는 도메인 이름이 필요합니다.
- HTTPS를 사용하여 웹 사이트가 작동해야 합니다. 이 사항은 보안 모범 사례이며 플러그인은 웹 사이트가 HTTPS를 사용하여 작동한다고 가정합니다. 확인하려면 HTTPS를 사용하는 웹 사이트 주소(예: https://example.com)로 이동하여 웹 사이트가 제대로 표시되는지 확인하십시오.

웹 사이트에 도메인 이름이 있고 웹 사이트가 HTTPS를 사용하여 작동하는 경우 다음 단원으로 진행하십시오.

1단계: 플러그인 설치

플러그인을 설치하기 전에 사전 조건 (p. 22)을 모두 갖추어야 합니다.

플러그인을 설치하려면

1. WordPress 웹 사이트 관리 대시보드(WP Admin)에 로그인합니다.
2. Plugins(플러그인)를 선택합니다.
3. - 이미 Amazon AI(플러그인의 이전 이름)나 AWS for WordPress 플러그인이 있는 경우:
     1. Amazon AI 또는 AWS for WordPress 옆의 확인란을 선택합니다.
     2. Bulk Action(대량 작업) 메뉴에서 업데이트를 선택한 후 적용을 선택합니다.
   - Amazon AI 또는 AWS for WordPress 플러그인이 없는 경우:
     1. 새로 추가를 선택합니다.
     2. 검색 상자에 **`AWS for WordPress`**를 입력합니다.

3. AWS for WordPress 플러그인을 찾습니다. Install Now(지금 설치)를 선택한 후 활성화를 선택합니다.

플러그인을 활성화한 후 다음 단원으로 진행하여 플러그인을 구성하고 사용합니다.

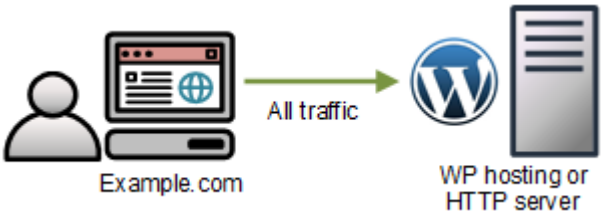
2단계: 플러그인으로 CloudFront 구성 및 사용

사이트 가속화를 위해 CloudFront를 AWS for WordPress 플러그인과 사용하면 플러그인이 하위 도메인(대체 도메인 이름 또는 CNAME)을 사용하여 CloudFront를 통해 웹 사이트의 트래픽을 전송합니다. 그러면 리소스를 보다 신속하게 로드하여 지연 시간을 줄이고 보다 나은 환경에서 콘텐츠를 볼 수 있습니다.

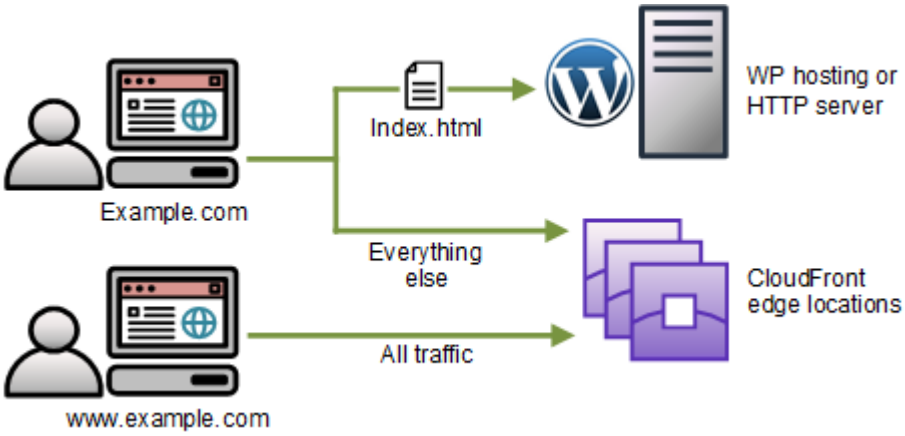
플러그인의 사이트 가속화 기능이 없으면 웹 사이트 최종 사용자의 모든 트래픽이 WordPress 웹 사이트를 호스팅하는 서버로 갑니다. 다음 절차의 단계를 완료한 후 플러그인의 사이트 가속화 기능을 활성화하여 최종 사용자에게 웹 사이트 방문 옵션 두 가지를 제시할 수 있습니다.

- 최종 사용자가 example.com과 같은 웹 사이트 도메인 이름을 사용하면 웹 사이트 인덱스 페이지와 작은 이미지 파일 몇 개를 제외한 모든 트래픽이 CloudFront를 통해 이동합니다.
- 최종 사용자가 www.example.com과 같은 웹 사이트 대체 도메인 이름을 사용하면 모든 트래픽이 CloudFront를 통해 이동합니다.

어느 도메인을 사용하건 지연 시간이 짧아지고 속도와 안정성이 향상되어 더 좋은 환경에서 웹 사이트를 볼 수 있습니다. 웹 사이트의 대체 도메인 이름을 사용하도록 최종 사용자에게 알려주시기 바랍니다. 다음 다이어그램에서는 플러그인의 사이트 가속화 사용 여부에 따른 최종 사용자의 트래픽을 보여줍니다.



플러그인의 사이트 가속화 사용 안 함



플러그인의 사이트 가속화 사용

플러그인을 사용하여 CloudFront를 구성하고 사용하려면(사이트 가속화 사용)

1. WordPress 웹 사이트 관리 대시보드(WP Admin)에 로그인합니다.
2. 왼쪽 탐색 창에서 AWS를 선택합니다.

3. 전에 저장한 (p. 22) 액세스 키 ID와 보안 액세스 키를 붙여 넣거나 입력한 후 변경 사항 저장을 선택합니다.

   Note

   Amazon EC2에서 WordPress를 호스팅하는 경우 이 단계를 건너뛰고 IAM 사용자 대신 IAM 역할을 사용할 수 있습니다. 이 경우 두 필드를 비워 두십시오. IAM 역할에 대한 자세한 내용은 Amazon EC2 사용 설명서의 Amazon EC2의 IAM 역할을 참조하십시오.

   Note

   AWS 리전 선택에 관계없이 플러그인의 CloudFront 기능이 미국 동부(버지니아 북부) 지역에 모든 리소스를 만듭니다.
4. 탐색 창에서 CloudFront를 선택합니다.
5. CloudFront Setup(CloudFront 설정) 페이지에서 다음을 수행합니다.
   a. 필요하면 Origin Domain Name(오리진 도메인 이름)에 웹 사이트의 도메인 이름을 입력합니다(예: example.com).
   b. CloudFront Alternate Domain Name(CloudFront 대체 도메인 이름)에 최종 사용자가 가속화된 웹 사이트 환경에 사용할 하위 도메인을 입력합니다. 웹 사이트 도메인 이름 앞에 www를 사용하는 것이 좋습니다(예: www.example.com).
   c. Initiate Setup(설정 시작)을 선택합니다.
6. CloudFront가 AWS Certificate Manager를 사용하여 대체 도메인 이름의 인증서를 만듭니다. 요청 후 72시간 안에 인증서를 확인해야 합니다. 플러그인의 설정 페이지에 표시되는 DNS 레코드를 추가하면 됩니다. 이 확인 레코드를 추가하는 프로세스는 DNS 서비스 공급자에 따라 다릅니다. WordPress.com 호스팅 서비스를 사용하는 경우 사용자 지정 항목으로 DNS 레코드를 업데이트하는 방법은 해당 설명서를 참조하십시오. DNS에 Amazon Route 53을 사용하는 경우 Amazon Route 53 개발자 안내서의 Amazon Route 53 콘솔을 사용하여 레코드 생성을 참조하십시오.

   DNS 레코드를 추가한 후 설정 페이지로 돌아가서 Check status of SSL certificate(SSL 인증서 상태 확인)를 선택합니다.

   이 단계를 완료하면 CloudFront에서 WordPress에 최적화된 배포를 설정합니다. 이 프로세스를 전역적으로 배포하는 데 시간이 좀 걸릴 수 있습니다. 설치 페이지가 10초마다 자동으로 새로 고쳐져 배포가 진행되는 동안 계속 업데이트됩니다.
7. 배포가 완료되면 DNS 레코드를 만들어 새 CloudFront 배포의 대체 도메인 이름(예: www.example.com)을 가리킵니다. 플러그인의 설정 페이지에 표시되는 DNS 레코드를 추가하면 됩니다. 이 확인 레코드를 추가하는 프로세스는 DNS 서비스 공급자에 따라 다릅니다. WordPress.com 호스팅 서비스를 사용하는 경우 사용자 지정 항목으로 DNS 레코드를 업데이트하는 방법은 해당 설명서를 참조하십시오. DNS에 Amazon Route 53을 사용하는 경우 Amazon Route 53 개발자 안내서의 Amazon Route 53 콘솔을 사용하여 레코드 생성을 참조하십시오.

   DNS 레코드를 추가한 후 설정 페이지로 돌아가 Check status of CloudFront DNS record(CloudFront DNS 레코드 상태 확인)를 선택합니다.
8. Activate Site Acceleration(사이트 가속화 활성화)을 선택한 후 변경 사항 저장을 선택합니다.

사이트 가속화를 활성화하면 AWS for WordPress 플러그인이 CloudFront 배포에서 웹 사이트의 리소스(예: CSS 및 JavaScript 파일, 이미지)를 서비스하도록 웹 사이트를 구성합니다. 프라이빗 검색 창에서 웹 사이트를 보거나 WordPress 관리 모드 밖의 다른 브라우저를 사용하여 플러그인이 최종 사용자를 위해 웹 사이트를 가속화하는지 확인할 수 있습니다. 대체 도메인 이름(예: www.example.com)을 사용하여 웹 사이트를 탐색할 수 있는지 확인합니다.

(선택 사항) 사이트 가속화 비활성화

사이트 가속화를 비활성화하여 CloudFront 배포를 거치지 않고 웹 서버 호스트에서 웹 사이트의 모든 리소스를 제공할 수 있습니다. 그러면 배포가 그대로 유지되어 플러그인의 사이트 가속화를 다시 활성화하도록 선택할 때 사용할 수 있습니다.

> **Warning**
>
> 사이트 가속화를 비활성화하기 전에 대체 도메인 이름(예: www.example.com)의 DNS 레코드를 편집하여 웹 사이트의 도메인을 가리키도록 합니다(예: example.com). 이 작업을 먼저 하지 않으면 웹 사이트에 가동 중지나 문제가 발생할 수 있습니다. DNS 레코드를 편집한 후 사이트 가속화를 비활성화하기 전에 레코드의 TTL(Time to Live) 값보다 오래 기다립니다.
> WordPress.com 호스팅 서비스를 사용하는 경우 DNS 레코드를 편집하는 방법은 해당 설명서를 참조하십시오. Amazon Route 53을 DNS에 사용하는 경우 Amazon Route 53 개발자 안내서의 레코드 편집을 참조하십시오.

사이트 가속화를 비활성화하려면

1. WordPress 웹 사이트에 로그인하고 WP Admin을 선택합니다.
2. 탐색 창에서 AWS를 선택합니다.
3. 탐색 창에서 CloudFront를 선택합니다.
4. Activate Site Acceleration(사이트 가속화 활성화) 확인란 선택을 취소하고 변경 사항 저장을 선택합니다.

사이트 가속화를 비활성화한 후 되돌릴 수 있습니다. 다시 활성화하려면 Activate Site Acceleration(사이트 가속화 활성화) 확인란을 선택한 후 변경 사항 저장을 선택합니다.

(선택 사항) 사이트 가속화 제거 및 CloudFront 배포 삭제

AWS for WordPress 플러그인을 사용하여 CloudFront 배포를 삭제할 수 있습니다. 삭제한 후 되돌릴 수 없습니다. AWS for WordPress 플러그인을 다시 사이트 가속화에 사용하려면 새 CloudFront 배포를 만들도록 플러그인을 다시 구성 (p. 25)해야 합니다.

> **Warning**
>
> CloudFront 배포를 삭제하기 전에, 웹 사이트의 도메인을 가리키도록 대체 도메인 이름(예: www.example.com)의 DNS 레코드를 편집합니다(예: example.com). 이 작업을 먼저 하지 않으면 웹 사이트에 가동 중지나 문제가 발생할 수 있습니다. DNS 레코드를 편집한 후 CloudFront 배포를 삭제하기 전에 레코드의 TTL(Time to Live) 값보다 오래 기다립니다.
> WordPress.com 호스팅 서비스를 사용하는 경우 DNS 레코드를 편집하는 방법은 해당 설명서를 참조하십시오. Amazon Route 53을 DNS에 사용하는 경우 Amazon Route 53 개발자 안내서의 레코드 편집을 참조하십시오.

사이트 가속화를 제거하고 CloudFront 배포를 삭제하려면

1. WordPress 웹 사이트에 로그인하고 WP Admin을 선택합니다.
2. 탐색 창에서 AWS를 선택합니다.
3. 탐색 창에서 CloudFront를 선택합니다.
4. Remove Site Acceleration(사이트 가속화 제거)을 선택한 후 확인을 선택합니다.

이 단계가 끝나면 AWS for WordPress 플러그인이 CloudFront 배포를 삭제합니다. 완료하는 데 몇 분 정도 걸릴 수 있습니다. 프로세스가 완료되면 필요에 따라 AWS Management 콘솔을 열어 플러그인이 만든 CloudFront, AWS Certificate Manager 및 AWS CloudFormation 리소스가 삭제되었는지 확인할 수 있습니다.

(선택 사항) 플러그인 비활성화 및 제거

AWS for WordPress 플러그인을 비활성화하여 CloudFront와 다른 AWS 서비스를 위한 모든 기능의 사용을 중지할 수 있습니다. 플러그인을 삭제하여 WordPress 웹 사이트에서 완전히 제거할 수도 있습니다.

> **Warning**
>
> 플러그인을 비활성화하고 삭제하기 전에 대체 도메인 이름(예: www.example.com)의 DNS 레코드를 편집하여 웹 사이트의 도메인을 가리키도록 합니다(예: example.com). 이 작업을 먼저 하지 않으면 웹 사이트에 가동 중지나 문제가 발생할 수 있습니다. DNS 레코드를 편집한 후 플러그인을 비활성화하고 삭제하기 전에 레코드의 TTL(Time to Live) 값보다 오래 기다립니다.
> WordPress.com 호스팅 서비스를 사용하는 경우 DNS 레코드를 편집하는 방법은 해당 설명서를 참조하십시오. Amazon Route 53을 DNS에 사용하는 경우 Amazon Route 53 개발자 안내서의 레코드 편집을 참조하십시오.

> **Note**
>
> 사이트 가속화를 먼저 제거하지 않고 플러그인을 비활성화 및 삭제하면 플러그인이 만든 CloudFront, AWS Certificate Manager 및 AWS CloudFormation 리소스가 삭제되지 않습니다. 이 리소스는 AWS 계정에 남아 있으며, AWS 프리 티어를 초과하는 사용량에 대해 요금이 청구됩니다. 플러그인을 삭제하기 전에 이 리소스를 삭제하려면 (선택 사항) 사이트 가속화 제거 및 CloudFront 배포 삭제 (p. 27) 단원을 참조하십시오.

AWS for WordPress 플러그인을 비활성화하려면

1. WordPress 웹 사이트에 로그인하고 WP Admin을 선택합니다.
2. Plugins(플러그인)를 선택합니다.
3. AWS for WordPress 플러그인을 찾은 후 비활성화를 선택합니다.

   플러그인을 비활성화한 후 되돌릴 수 있습니다. 다시 활성화하려면 활성화를 선택합니다.
4. AWS for WordPress 플러그인을 완전히 제거하려면 삭제를 선택합니다.

(선택 사항) Amazon Polly 콘텐츠용 CloudFront 배포 만들기

AWS for WordPress 플러그인을 Amazon Polly와 함께 사용하면 CloudFront 배포를 만들어 Amazon Polly에서 생성된 오디오 콘텐츠를 가속화할 수 있습니다. Amazon Polly와 함께 플러그인을 사용하는 방법에 대한 자세한 내용은 Amazon Polly 개발자 안내서의 Amazon Polly용 WordPress 플러그인을 참조하십시오.

Amazon Polly 오디오용 CloudFront 배포를 만들려면

1. WordPress 웹 사이트 관리 대시보드(WP Admin)에 로그인합니다.
2. 왼쪽 탐색 창에서 AWS를 선택합니다.
3. 클라우드 스토리지 섹션에서 S3 버킷 이름을 적어 둡니다. audio-for-wordpress 또는 audio_for_wordpress로 시작됩니다. 다음 단계를 완료하려면 이 버킷 이름이 필요합니다.
4. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
5. [Create Distribution]을 선택합니다.
6. Web(웹) 배포에서 시작하기를 선택합니다.
7. Origin Domain Name(오리진 도메인 이름)에서 전 단계에서 이름을 적어 놓은 Amazon S3 버킷을 선택합니다.
8. 페이지 하단으로 스크롤한 다음 Create Distribution(배포 생성)을 선택합니다.

9. 전 단계에서 만든 배포를 선택한 후 배포의 도메인 이름을 적어 둡니다. 다음 단계를 완료하려면 이 도메인 이름이 필요합니다.
10. WordPress 웹 사이트 관리 대시보드(WP Admin)에 로그인합니다.
11. 왼쪽 탐색 창에서 AWS를 선택합니다.
12. Amazon CloudFront (CDN) domain name(Amazon CloudFront(CDN) 도메인 이름)에 전 단계에서 적어 놓은 도메인 이름을 입력합니다.
13. 변경 사항 저장을 선택합니다.

문제 해결

AWS for WordPress 플러그인에 문제가 발생한다면 다음 주제를 참고하여 해결할 수 있습니다. 버그를 보고하거나 이 주제에서 다루지 않은 다른 문제에 대한 도움을 받으려면 GitHub에 문제를 게시하십시오.

주제

- AWS에 연결할 수 없음 (p. 29)
- 사용자에게 권한이 없습니다. (p. 29)
- CloudFront 설정 페이지가 비어 있음 (p. 29)
- DescribeCertificate 오류 (p. 30)
- AWS CloudFormation 오류 (p. 30)
- CloudFront 배포 작업이 중단된 것 같음 (p. 30)
- 대체 도메인이 작동하지 않음 (p. 30)
- AWS 리소스를 찾을 수 없음 (p. 30)

AWS에 연결할 수 없음

플러그인에 AWS에 연결할 수 없습니다. 자격 증명을 확인하고 AWS 계정이 활성 상태인지 확인하십시오. 오류가 표시될 수 있습니다. 이 오류가 표시되면 다음을 시도해 보십시오.

- 플러그인의 General configuration(일반 구성) 페이지에 AWS 액세스 키와 AWS 보안 키를 입력했는지 확인합니다. 자세한 내용은 2단계: 플러그인으로 CloudFront 구성 및 사용 (p. 25) 단원을 참조하십시오.
- 플러그인용으로 만든 IAM 사용자에게 올바른 권한이 있는지 확인합니다. 자세한 내용은 IAM 사용자 생성 (p. 23) 단원을 참조하십시오.

사용자에게 권한이 없습니다.

플러그인에 다음과 같은 오류 메시지가 표시될 수 있습니다.

- 설치 중 오류
- AccessDenied
- 사용자: `<### ARN>`이 `<##>`을 수행할 권한이 없습니다.

이 오류 중 하나가 표시되면 플러그인용으로 만든 IAM 사용자에게 올바른 권한이 있는지 확인하십시오. 자세한 내용은 IAM 사용자 생성 (p. 23) 단원을 참조하십시오.

CloudFront 설정 페이지가 비어 있음

플러그인의 CloudFront 설정 페이지로 이동했는데 페이지가 비어 있을 수 있습니다. 플러그인의 General configuration(일반 구성) 페이지에 AWS 액세스 키와 AWS 보안 키를 입력하지 않은 것입니다. 자세한 내용은 2단계: 플러그인으로 CloudFront 구성 및 사용 (p. 25) 단원을 참조하십시오.

DescribeCertificate 오류

플러그인에 다음과 같은 오류 메시지가 표시될 수 있습니다.

- 설치 중 오류
- DescribeCertificate 작업을 위해 제공된 입력을 확인하는 동안 오류 1개를 발견했습니다. [CertificateArn]에 필요한 문자열 길이가 20자 이상인데 발견된 문자열 길이는 0입니다.

이 오류 중 하나가 표시되면 Restart Setup(설정 다시 시작)을 선택한 후 Origin Domain Name(오리진 도메인 이름)과 CloudFront Alternate Domain Name(CloudFront 대체 도메인 이름)에 IP 주소가 아닌 도메인 이름을 입력해야 합니다. 자세한 내용은 2단계: 플러그인으로 CloudFront 구성 및 사용 (p. 25) 단원을 참조하십시오.

AWS CloudFormation 오류

플러그인에 다음과 같은 오류 메시지가 표시될 수 있습니다.

- 메서드 AmazonAI_Cloudformation에 예외가 발생했습니다.
- 스택이 예기치 않은 상태입니다. CloudFront 배포 상태가 <## ##>이고 스택 상태가 ## ##입니다.

이 오류 중 하나가 표시되면 Restart Setup(설정 다시 시작)을 선택하여 다시 시도하십시오. AWS CloudFormation 콘솔을 사용하여 편안하게 오류를 진단하는 경우 콘솔을 열어 무엇이 잘못되었는지 확인할 수 있습니다.

CloudFront 배포 작업이 중단된 것 같음

사이트 가속화를 설정할 때 플러그인이 CloudFront Distribution Deployment(CloudFront 배포 작업) 단계를 오래 표시하기도 하는데 이 단계에서 플러그인이 중단된 것처럼 보입니다. 이 단계를 완료되는 데 몇 분 정도 걸릴 수 있습니다. 이 단계에서 플러그인이 10초마다 새로 고쳐지고 <## #### ## # ##># ## #####과 같은 메시지가 표시됩니다. 플러그인이 페이지를 마지막으로 새로 고친 시간을 확인하려면 이 메시지를 찾아보십시오. 이 시간에서 1분이 채 지나지 않았다면 이 단계가 완료될 때까지 계속 기다리시기 바랍니다. 플러그인이 한동안 새로 고쳐지지 않았으면 페이지를 다시 로드해도 좋습니다.

대체 도메인이 작동하지 않음

플러그인으로 CloudFront 설정을 끝냈지만 대체 도메인 이름(예: www.example.com)이 작동하지 않으면 CNAME 레코드를 DNS 레코드에 추가했는지 확인하십시오. WordPress.com 호스팅 서비스를 사용하는 경우 사용자 지정 항목으로 DNS 레코드를 업데이트하는 방법은 해당 설명서를 참조하십시오. DNS에 Amazon Route 53을 사용하는 경우 Amazon Route 53 개발자 안내서의 Amazon Route 53 콘솔을 사용하여 레코드 생성을 참조하십시오.

플러그인을 사용하여 CloudFront 배포를 삭제한 경우 대체 도메인 이름(예: www.example.com)이 작동하지 않으면 대체 도메인 이름이 웹 사이트의 apex 도메인(예: example.com)을 다시 가리키도록 DNS 레코드를 업데이트했으며 DNS 레코드의 TTL(Time to Live) 값인지 확인하십시오. WordPress.com 호스팅 서비스를 사용하는 경우 사용자 지정 항목으로 DNS 레코드를 업데이트하는 방법은 해당 설명서를 참조하십시오. DNS에 Amazon Route 53을 사용하는 경우 Amazon Route 53 개발자 안내서의 Amazon Route 53 콘솔을 사용하여 레코드 생성을 참조하십시오.

AWS 리소스를 찾을 수 없음

플러그인의 CloudFront 기능은 CloudFront, AWS Certificate Manager 및 AWS CloudFormation을 비롯하여 여러 AWS 서비스에서 리소스를 만듭니다. AWS Management 콘솔에서 이 리소스를 찾거나 API를 사용하여 나열하려면 미국 동부(버지니아 북부) 지역(us-east-1)을 사용해야 합니다.

배포 작업

CloudFront 배포를 생성하여 CloudFront에게 어디로부터 콘텐츠를 전송하고자 하는지와 이러한 콘텐츠 전송을 추적 및 관리하는 방법에 대한 세부 정보를 알립니다. 다음 항목에서는 CloudFront 배포에 대한 몇 가지 기본 내용을 설명하고, 해당 비즈니스 요건에 맞게 배포를 구성하기 위해 선택할 수 있는 설정에 대한 자세한 정보를 제공합니다.

> **Note**
>
> Adobe Flash Media Server의 RTMP 프로토콜을 사용하는 경우 다른 종류의 CloudFront 배포를 설정합니다. 자세한 내용은 RTMP 배포 작업 (p. 268) 단원을 참조하십시오.

주제

- 배포의 개요 (p. 31)
- 배포 만들기, 업데이트, 삭제 (p. 33)
- 웹 배포를 위해 Amazon S3 오리진, MediaPackage 채널, 사용자 지정 오리진 사용 (p. 61)
- 대체 도메인 이름(CNAME)을 추가하여 파일에 대해 사용자 지정 URL 사용 (p. 66)
- CloudFront 배포를 통한 WebSocket 사용 (p. 75)

배포의 개요

CloudFront를 사용하여 콘텐츠를 배포하려는 경우 배포를 생성하고 원하는 구성 설정을 선택합니다. 예:

- 콘텐츠 오리진 - CloudFront가 배포할 파일을 가져오는 Amazon S3 버킷, MediaPackage 채널 또는 HTTP 서버. 최대 25개의 Amazon S3 버킷, 채널 및/또는 HTTP 서버를 결합하여 오리진으로 지정할 수 있습니다.
- 액세스 - 파일을 모든 사람이 사용할 수 있도록 할지 아니면 일부 사용자로만 액세스를 제한할지 여부.
- 보안 - CloudFront에서 사용자가 HTTPS를 사용하여 콘텐츠에 액세스하도록 지정할지 여부.
- 쿠키 또는 쿼리 문자열 전송 - CloudFront에서 쿠키 및/또는 쿼리 문자열을 오리진으로 전달할지 여부.
- 지리적 제약 - CloudFront에서 특정 국가의 사용자가 콘텐츠에 액세스하는 것을 차단할지 여부.
- 액세스 로그 - CloudFront가 최종 사용자 활동을 표시하는 액세스 로그를 생성할지 여부.

현재 각 AWS 계정에 대해 생성할 수 있는 배포의 최대 수는 웹 배포의 일반 할당량 (p. 447) 및 RTMP 배포에 대한 할당량 (p. 452) 단원을 참조하십시오. 배포당 제공할 수 있는 최대 파일 수는 없습니다.

배포를 사용하여 HTTP 또는 HTTPS를 통해 다음 콘텐츠를 제공할 수 있습니다.

- HTTP 또는 HTTPS를 사용하는 정적 및 동적 다운로드 콘텐츠입니다(예: .html, .css, .js 및 이미지 파일).
- Apple HTTP Live Streaming(HLS) 및 Microsoft Smooth Streaming 등과 같은 다양한 형식의 온디맨드 비디오. 자세한 내용은 CloudFront를 사용한 온디맨드 비디오(VOD) 제공 (p. 262) 단원을 참조하십시오.

  Adobe Flash 멀미미디어는 HTTP 또는 HTTPS를 통해 제공할 수 없지만, CloudFront RTMP 배포를 사용하여 제공할 수는 있습니다. RTMP 배포 (p. 269) 단원을 참조하십시오.
- 실시간으로 발생하는 모임, 회의 또는 콘서트 같은 라이브 이벤트 제공. 라이브 스트리밍의 경우 AWS CloudFormation 스택을 사용하여 배포를 자동으로 생성할 수 있습니다. 자세한 내용은 CloudFront 및 AWS Media Services를 사용하여 라이브 스트리밍 비디오 제공 (p. 263) 단원을 참조하십시오.

배포 만들기에 대한 자세한 내용은 배포 만들기 단계(개요) (p. 34) 단원을 참조하십시오.

배포 시 사용할 수 있는 작업

다음 표에서는 배포 작업을 위해 선택할 수 있는 CloudFront 작업을 나열하고 CloudFront 콘솔 및 CloudFront API를 사용하여 해당 작업을 수행하는 방법에 대한 해당 설명서의 링크를 제공합니다.

작업	CloudFront 콘솔 사용	CloudFront API: 웹 배포 사용	CloudFront API: RTMP 배포 사용
배포 만들기	웹 배포: 배포 만들기 단계(개요) (p. 34) 단원 참조 RTMP 배포: RTMP를 사용한 미디어 파일 스트리밍용 작업 목록 (p. 271) 단원 참조	CreateDistribution으로 이동	CreateStreamingDistribution으로 이동
배포 목록 보기	배포 업데이트 (p. 59) 단원을 참조하십시오.	ListDistributions로 이동	ListStreamingDistributions로 이동
배포에 대한 모든 정보 보기	배포 업데이트 (p. 59) 단원을 참조하십시오.	GetDistribution으로 이동	GetStreamingDistribution으로 이동
배포 구성 가져오기	배포 업데이트 (p. 59) 단원을 참조하십시오.	GetDistributionConfig로 이동	GetStreamingDistributionConfig로 이동
배포 업데이트	배포 업데이트 (p. 59) 단원을 참조하십시오.	UpdateDistribution으로 이동	UpdateStreamingDistribution으로 이동
배포 삭제	배포 삭제 (p. 60) 단원을 참조하십시오.	DeleteDistribution으로 이동	DeleteStreamingDistribution으로 이동

배포 생성 및 배포 업데이트에 필요한 필드

UpdateDistribution CloudFront API 작업을 사용해 배포를 업데이트할 때에는 CreateDistribution을 사용하여 배포를 생성할 때보다 필수 필드가 더 많습니다. 배포 만들기 및 업데이트에 필요한 필드에 대한 요약 설명은 아래 표를 참조하십시오.

DistributionConfig

회원	API 호출을 만들 때 필요한지 여부	API 호출을 업데이트할 때 필요한지 여부
CallerReference	Y	Y
별칭	-	Y
DefaultRootObject	-	Y
오리진	Y	Y
DefaultCacheBehavior	Y	Y
CacheBehaviors	-	Y

회원	API 호출을 만들 때 필요한지 여부	API 호출을 업데이트할 때 필요한지 여부
CustomErrorResponses	-	Y
Comment	Y	Y
로깅	-	Y
PriceClass	-	Y
활성	Y	Y
ViewerCertificate	-	Y
제한 사항	-	Y
WebACLId	-	Y
HttpVersion	-	Y
IsIPV6Enabled	-	-

CacheBehavior

회원	API 호출을 만들 때 필요한지 여부	API 호출을 업데이트할 때 필요한지 여부
PathPattern	Y	Y
TargetOriginId	Y	Y
ForwardedValues	Y	Y
TrustedSigners	Y	Y
ViewerProtocolPolicy	Y	Y
MinTTL	Y	Y
AllowedMethods	-	Y
SmoothStreaming	-	Y
DefaultTTL	-	Y
MaxTTL	Y	Y
압축	-	Y
LambdaFunctionAssociations	-	Y
FieldLevelEncryptionId	-	Y

배포 만들기, 업데이트, 삭제

다음 항목의 단계를 완료하여 배포를 생성하거나 업데이트하거나 삭제할 수 있습니다.

주제

- 배포 만들기 단계(개요) (p. 34)
- 배포 만들기 (p. 35)
- 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35)
- CloudFront가 콘솔에 표시하는 값 (p. 58)
- 배포 테스트 (p. 59)
- 배포 업데이트 (p. 59)
- 배포 삭제 (p. 60)

배포 만들기 단계(개요)

다음 작업 목록은 배포를 생성하기 위한 프로세스를 요약하여 보여 줍니다.

배포를 생성하려면

1. 하나 이상의 Amazon S3 버킷을 만들거나 HTTP 서버를 오리진 서버로 구성합니다. 오리진은 콘텐츠의 원본 버전을 저장하는 장소입니다. CloudFront에서 파일에 대한 요청을 가져오면, 이 요청은 엣지 로케이션에서 배포할 파일을 가져오는 오리진이 됩니다. Amazon S3 버킷 및 HTTP 서버를 결합하여 오리진 서버로 사용할 수 있습니다.

   Amazon S3를 사용 중인 경우 버킷의 이름은 전부 소문자로만 구성되어야 하며 공백을 포함할 수 없습니다.

   Amazon EC2 서버나 다른 사용자 지정 서버를 사용 중인 경우, Amazon EC2 또는 기타 사용자 지정 오리진 사용 (p. 63) 단원을 검토하십시오.

   현재 하나의 배포에 대해 생성할 수 있는 오리진의 최대 수를 살펴보고 더 높은 할당량(이전에는 제한이라고 함)을 요청하려면, 웹 배포의 일반 할당량 (p. 447) 단원을 참조하십시오.
2. 오리진 서버에 콘텐츠를 업로드합니다. CloudFront 서명된 URL을 사용한 콘텐츠에 대한 액세스 제한을 원치 않는 경우, 객체를 공개적으로 읽기 가능하도록 설정합니다.

   > **Important**
   >
   > 이 경우, 오리진 서버의 보안을 보장하는 것은 사용자의 책임입니다. CloudFront에서 서버에 대한 액세스 권한을 보유해야 하며 콘텐츠를 안전하게 보호할 수 있도록 적절한 보안 설정이 적용되어 있어야 합니다.
3. CloudFront 배포를 만듭니다.

   - CloudFront 콘솔을 사용하여 배포를 만드는 자세한 내용은 배포 만들기 (p. 35) 단원을 참조하십시오.
   - CloudFront API를 사용한 배포 만들기에 대한 자세한 내용은 Amazon CloudFront API Reference의 CreateDistribution 단원을 참조하십시오.
4. 선택 사항: CloudFront 콘솔을 사용하여 배포를 만든 경우, 캐시 동작 또는 배포에 대한 오리진을 추가로 만듭니다. 자세한 내용은 CloudFront 배포를 업데이트하려면 (p. 59) 단원을 참조하십시오.
5. 배포를 테스트합니다. 자세한 내용은 배포 테스트 (p. 59) 단원을 참조하십시오.
6. 3단계에서 배포를 만든 뒤 CloudFront에서 반환된 도메인 이름을 사용하여 콘텐츠에 액세스하는 웹 사이트 또는 애플리케이션을 작성합니다. 예를 들어 CloudFront에서 d111111abcdef8.cloudfront.net을 배포의 도메인 이름으로 반환할 경우, Amazon S3 버킷 또는 HTTP 서버의 루트 디렉터리에 있는 `image.jpg` 파일의 URL은 `http://d111111abcdef8.cloudfront.net/image.jpg`가 됩니다.

   배포를 만들 때 하나 이상의 대체 도메인 이름(CNAME)을 지정한 경우, 자체 도메인 이름을 사용할 수 있습니다. 이 경우 `image.jpg`에 대한 URL은 `http://www.example.com/image.jpg`가 될 수 있습니다.

   다음을 참조하십시오.

- 서명된 URL을 사용하여 콘텐츠에 대한 액세스를 제한하려는 경우 서명된 URL과 서명된 쿠키를 사용하여 프라이빗 콘텐츠 제공 (p. 115) 단원을 참조하십시오.
- 압축된 콘텐츠를 제공하려는 경우 압축된 파일 제공 (p. 87) 단원을 참조하십시오.
- Amazon S3 및 사용자 지정 오리진에 대한 CloudFront 요청 및 응답 동작과 관련한 자세한 내용은 요청 및 응답 동작 (p. 230) 단원을 참조하십시오.

배포 만들기

CloudFront 콘솔을 사용하거나 프로그래밍 방식으로 배포를 만들거나 업데이트할 수 있습니다. 이 항목에서는 콘솔을 사용한 배포 작업을 설명합니다.

CloudFront API를 사용하여 배포를 만들거나 업데이트하려면 Amazon CloudFront API Reference의 배포 만들기 또는 배포 업데이트 단원을 참조하십시오.

> **Important**
>
> 배포를 업데이트하는 경우에는 배포를 만들 때는 필요 없었던 여러 개의 추가 필드가 필요합니다. CloudFront API를 사용하여 배포를 업데이트하는 경우 필수 필드를 모두 포함하려면 Amazon CloudFront API Reference에 설명된 단계를 따르십시오.

현재 각 AWS 계정에 대해 생성할 수 있는 배포의 최대 수를 살펴보고 더 높은 할당량(이전에는 제한이라고 함)을 요청하려면, 웹 배포의 일반 할당량 (p. 447) 단원을 참조하십시오.

CloudFront 웹 배포를 만들려면(콘솔)

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. [Create Distribution]을 선택합니다.
3. Create Distribution Wizard(배포 만들기 마법사)의 첫 번째 페이지에서 Web(웹) 섹션의 Get Started(시작하기)를 선택합니다.
4. 배포에 대해 설정을 지정합니다. 자세한 내용은 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.
5. 변경 내용을 저장합니다.
6. CloudFront에서 배포를 생성하면 배포에 대한 상태 열의 값이 진행 중에서 배포 완료로 변경됩니다. 배포를 활성화하도록 선택한 경우 상태가 배포 완료로 전환되면 요청을 처리할 수 있습니다.

   CloudFront가 배포에 할당하는 도메인 이름이 배포 목록에 나타납니다. 선택한 배포에 대한 [General] 탭에도 나타납니다.

   > **Tip**
   >
   > 대체 도메인 이름(CNAME)을 추가하여 파일에 대해 사용자 지정 URL 사용 (p. 66)의 단계를 따르면 CloudFront;에서 할당한 이름 대신 대체 도메인 이름을 사용할 수 있습니다.
7. 배포가 완료되면 새 CloudFront URL 또는 CNAME을 사용하여 콘텐츠에 액세스할 수 있는지 확인합니다. 자세한 내용은 배포 테스트 (p. 59) 단원을 참조하십시오.

배포를 업데이트하려면(예: 캐시 동작 추가 또는 변경) 배포 업데이트 (p. 59) 단원을 참조하십시오.

배포를 만들거나 업데이트할 때 지정하는 값

배포를 새로 만들거나 기존 배포를 업데이트할 경우 다음 값을 지정합니다. CloudFront 콘솔을 사용하여 배포를 만들거나 업데이트하는 자세한 내용은 배포 만들기 (p. 35) 또는 배포 업데이트 (p. 59) 단원을 참조하십시오.

제공 형태 (p. 37)

제공 형태: 웹 또는 RTMP를 지정합니다. 자세한 정보는 제공 형태 (p. 37) 단원을 참조하십시오.

오리진 설정 (p. 38)

- Origin Domain Name (p. 38)
- Origin Path (p. 39)
- Origin ID (p. 39)

다음 값은 Amazon S3 오리진에서만 사용할 수 있습니다.

- 버킷 액세스 제한 (p. 40)
- 원본 액세스 ID (p. 40)
- 새 ID 설명 (p. 40)
- 사용자 ID (p. 40)
- 버킷에 대한 읽기 권한 부여 (p. 40)

다음 값은 사용자 지정 오리진(예: Amazon EC2, Elastic Load Balancing, 웹 사이트 엔드포인트로 구성된 Amazon S3 버킷, 자체 웹 서버)에서만 사용할 수 있습니다.

- 최소 오리진 SSL 프로토콜 (p. 41)
- 오리진 프로토콜 정책 (p. 41)
- 오리진 응답 제한 시간 (p. 42)
- 오리진 연결 유지 제한 시간 (p. 42)
- HTTP 포트 (p. 42)
- HTTPS 포트 (p. 43)

다음 값은 모든 유형의 오리진에서 사용할 수 있습니다.

- 오리진 사용자 지정 헤더 (p. 43)

캐시 동작 설정 (p. 43)

- 경로 패턴 (p. 44)
- Origin(기존 배포에만 해당) (p. 45)
- Viewer Protocol Policy (p. 45)
- Allowed HTTP Methods (p. 46)
- 필드 레벨 암호화 (p. 45)
- Cached HTTP Methods (p. 46)
- 선택한 요청 헤더 기반의 캐시 (p. 46)
- Whitelist Headers (p. 47)
- Object Caching (p. 47)
- Minimum TTL (p. 47)
- Maximum TTL (p. 47)
- 기본 TTL (p. 48)
- 쿠키 전달 (p. 48)
- 화이트리스트 쿠키 (p. 48)

- Query String Forwarding and Caching (p. 49)
- Query String Whitelist (p. 49)
- Smooth Streaming (p. 49)
- Restrict Viewer Access (Use Signed URLs) (p. 49)
- Trusted Signers (p. 50)
- AWS Account Numbers (p. 50)
- 자동으로 객체를 압축 (p. 50)
- 이벤트 유형 (p. 50)
- Lambda함수 ARN (p. 51)

배포 세부 정보 (p. 51)

- 요금 계층 (p. 51)
- AWS WAF 웹 ACL (p. 51)
- Alternate Domain Names (CNAMEs) (p. 51)
- SSL 인증서 (p. 52)
- Clients Supported (p. 52)
- 보안 정책 (p. 53)
- 최소 SSL 보안 프로토콜 – 보안 정책 (p. 53) 참조
- Supported HTTP Versions (p. 54)
- 기본 루트 객체 (p. 54)
- 로깅 (p. 54)
- Bucket for Logs (p. 54)
- Log Prefix (p. 55)
- Cookie Logging (p. 55)
- IPv6 사용 (p. 55)
- 의견 (p. 56)
- Distribution State (p. 56)

사용자 지정 오류 페이지 및 오류 캐싱 (p. 56)

- Error Code (p. 56)
- Response Page Path (p. 57)
- Response Code (p. 57)
- Error Caching Minimum TTL (p. 57)

제한 사항 (p. 57)

- Enable Geo Restriction (p. 57)
- Restriction Type (p. 57)
- Countries (p. 57)

제공 형태

배포를 만들 때는 전달 방법을 지정합니다. Adobe Flash Media Server와 RTMP를 사용하지 않은 경우 이 값은 항상 웹입니다. 기존 배포에 대한 전달 방법은 변경할 수 없습니다.

오리진 설정

배포를 생성 또는 업데이트할 경우, 오리진이라고도 하는 원래 버전의 웹 콘텐츠를 저장하는 하나 이상의 위치에 대한 정보를 제공합니다. CloudFront는 오리진에서 웹 콘텐츠를 가져와 이를 엣지 서버의 전 세계 네트워크를 통해 최종 사용자에게 제공합니다. 각 오리진은 Amazon S3 버킷 또는 HTTP 서버(예: 웹 서버) 중 하나입니다.

현재 하나의 배포에 대해 생성할 수 있는 오리진의 최대 수를 살펴보고 더 높은 할당량(이전에는 제한이라고 함)을 요청하려면, 웹 배포의 일반 할당량 (p. 447) 단원을 참조하십시오.

오리진을 삭제하려는 경우 먼저 해당 오리진에 연결된 캐시 동작을 편집하거나 삭제해야 합니다.

> **Important**
>
> 오리진을 삭제할 경우 이전에 해당 오리진에서 제공된 파일이 다른 오리진에서 사용 가능하고 캐시 동작이 현재 그러한 파일에 대한 요청을 새 오리진에 라우팅하는지 확인합니다.

배포를 생성 또는 업데이트할 경우 각 오리진에 대해 다음 값을 지정합니다.

Origin Domain Name

CloudFront에서 이 오리진에 대한 객체를 가져오려는 대상 Amazon S3 버킷 또는 HTTP 서버의 DNS 도메인 이름으로 다음과 같습니다.

- Amazon S3 버킷 – `awsexamplebucket.s3.us-west-2.amazonaws.com`

  > **Note**
  >
  > 최근에 S3 버킷을 만든 경우 CloudFront 배포에서 최대 24시간 동안 `HTTP 307 Temporary Redirect` 응답을 반환할 수 있습니다. S3 버킷 이름이 모든 AWS 리전에 전파되는 데 최대 24시간이 걸릴 수 있습니다. 전파가 완료되면, 배포에서는 이러한 리디렉션 응답 전송을 자동으로 중지하므로 별도로 조치를 취할 필요가 없습니다. 자세한 내용은 Amazon S3에서 HTTP 307 임시 리디렉션 응답을 받는 이유는 무엇입니까? 및 임시 요청 리디렉션을 참조하십시오.
- 웹 사이트로 구성된 Amazon S3 버킷 – `https://awsexamplebucket.s3-website.us-west-2.amazonaws.com`
- MediaStore 컨테이너 – `examplemediastore.data.mediastore.us-west-1.amazonaws.com`
- MediaPackage 엔드포인트 – `examplemediapackage.mediapackage.us-west-1.amazonaws.com`
- Amazon EC2 인스턴스 – `ec2-203-0-113-25.compute-1.amazonaws.com`
- Elastic Load Balancing 로드 밸런서 – `example-load-balancer-1234567890.us-west-2.elb.amazonaws.com`
- 자체 웹 서버 – `https://example.com`

Origin Domain Name(오리진 도메인 이름) 필드에서 도메인 이름을 선택하거나, 이름을 입력합니다. 도메인 이름은 대/소문자를 구분하지 않습니다.

오리진이 Amazon S3 버킷인 경우 다음에 유의하십시오.

- 버킷이 웹 사이트로 구성된 경우, 버킷에 대한 Amazon S3 정적 웹 사이트 호스팅 엔드포인트를 입력합니다. Origin Domain Name(오리진 도메인 이름) 필드의 목록에서 버킷 이름을 선택하지 마십시오. 정적 웹 사이트 호스팅 엔드포인트는 Amazon S3 콘솔에서 속성 페이지의 정적 웹 사이트 호스팅 아래에 나타납니다. 자세한 내용은 오리진에 대한 웹 사이트 엔드포인트로 구성된 Amazon S3 버킷 사용 (p. 62) 단원을 참조하십시오.
- 버킷에 대해 Amazon S3 Transfer Acceleration을 구성한 경우, Origin Domain Name(오리진 도메인 이름)에 `s3-accelerate` 엔드포인트를 지정하지 마십시오.
- 다른 AWS 계정에서 버킷을 사용하려는 경우 버킷이 웹 사이트로 구성되어 있지 않으면 다음 형식을 사용하여 이름을 입력합니다.

```
bucket-name.s3.region.amazonaws.com
```

버킷이 미국 표준 리전에 있고 Amazon S3에서 요청을 버지니아 북부의 시설로 라우팅하려는 경우, 다음 형식을 사용합니다.

```
bucket-name.s3.us-east-1.amazonaws.com
```

- CloudFront 원본 액세스 ID를 사용하여 Amazon S3의 콘텐츠에 보안을 적용하지 않는 한, 파일은 공개적으로 읽기 가능해야 합니다. 자세한 내용은 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오.

  Important

  오리진이 Amazon S3 버킷인 경우, DNS 명명 요구 사항을 준수하는 버킷 이름을 사용해야 합니다. 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 버킷 규제 및 제한 단원을 참조하십시오.

오리진에 대한 Origin Domain Name(오리진 도메인 이름)의 값을 변경하면 CloudFront에서는 CloudFront 엣지 로케이션에 대한 변경 사항 복제를 즉시 시작합니다. 배포 구성이 지정된 엣지 로케이션에서 업데이트될 때까지 CloudFront에서는 계속해서 요청을 이전 HTTP 서버 또는 Amazon S3 버킷으로 전달합니다. 배포 구성이 해당 엣지 로케이션에서 업데이트되는 즉시 CloudFront에서는 요청을 새 HTTP 서버 또는 Amazon S3 버킷으로 전달하기 시작합니다.

오리진을 변경하더라도 CloudFront에서는 엣지 캐시를 새 오리진의 객체로 다시 채울 필요가 없습니다. 애플리케이션의 최종 사용자 요청이 변경되지 않는 한, CloudFront에서는 각 객체의 TTL이 만료되거나 자주 요청되지 않는 객체가 제거될 때까지는 계속해서 엣지 캐시에 이미 존재하는 객체를 제공합니다.

Origin Path

CloudFront가 해당 AWS 리소스나 사용자 지정 오리진에 있는 디렉터리에서 콘텐츠를 요청하게 하려면 슬래시(/) 뒤에 디렉터리 경로를 입력합니다. CloudFront에서는 Origin Domain Name(오리진 도메인 이름) 값에 디렉터리 경로를 추가합니다(예: cf-origin.example.com/production/images). 경로 끝에는 슬래시(/)를 추가하지 마십시오.

예를 들어, 배포에 대해 다음 값을 지정했다고 가정합니다.

- Origin Domain Name(오리진 도메인 이름) – 이름이 myawsbucket인 Amazon S3 버킷
- Origin Path(오리진 경로) – /production
- Alternate Domain Names (CNAMEs)대체 도메인 이름(CNAME) – example.com

사용자가 브라우저에 example.com/index.html을 입력하면 CloudFront에서는 Amazon S3에 myawsbucket/production/index.html에 대한 요청을 보냅니다.

한편, 사용자가 브라우저에 example.com/acme/index.html을 입력하면 CloudFront에서는 Amazon S3에 myawsbucket/production/acme/index.html에 대한 요청을 보냅니다.

Origin ID

이 배포 내의 이 오리진 또는 오리진 그룹을 고유하게 구별하는 문자열입니다. 기본 캐시 동작에 추가로 캐시 동작을 만들 경우, 여기에서 지정한 ID를 사용하여 요청이 해당 캐시 동작의 경로 패턴과 일치할 때 CloudFront에서 요청을 라우팅하려는 오리진 또는 오리진 그룹을 식별합니다.

자세한 정보는 다음 단원을 참조하십시오.

- 지정할 수 있는 오리진: CloudFront 오리진 그룹 사용 (p. 64)

- 오리진 그룹 만들기: 오리진 그룹 만들기 (p. 212)
- 캐시 동작으로 작업: 캐시 동작 설정 (p. 43)

버킷 액세스 제한

> Note
>
> Amazon S3 버킷 오리진에만 적용됩니다(웹 사이트 엔드포인트로 구성된 경우는 제외).

사용자에게 Amazon S3 URL이 아닌 CloudFront URL만 사용하여 Amazon S3 버킷의 객체에 액세스하게 하려면 예를 선택합니다. 그런 다음 추가 값을 지정합니다.

사용자가 CloudFront URL 또는 Amazon S3 URL을 사용하여 객체에 액세스할 수 있도록 하려면 아니요를 선택합니다.

자세한 정보는 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오.

사용자가 CloudFront URL만 사용하여 사용자 지정 오리진의 객체에 액세스할 수 있도록 요구하는 방법에 대한 자세한 내용은 사용자 지정 오리진의 파일에 대한 액세스 제한 (p. 117) 단원을 참조하십시오.

원본 액세스 ID

> Note
>
> Amazon S3 버킷 오리진에만 적용됩니다(웹 사이트 엔드포인트로 구성된 경우는 제외).

Restrict Bucket Access(버킷 액세스 제한)에 대해 예를 선택한 경우, 새로운 원본 액세스 ID를 만들지 아니면 AWS 계정에 연결된 기존 원본 액세스 ID를 사용할지를 선택합니다. 원본 액세스 ID가 이미 있는 경우 이것을 재사용하여 유지 보수를 단순화하는 것이 좋습니다. 원본 액세스 ID에 대한 자세한 내용은 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오.

새 ID 설명

> Note
>
> Amazon S3 버킷 오리진에만 적용됩니다(웹 사이트 엔드포인트로 구성된 경우는 제외).

Origin Access Identity(원본 액세스 ID)에 대해 Create a New Identity(새 자격 증명 생성)를 선택한 경우, 새 원본 액세스 ID를 식별하는 설명을 입력합니다. 이 배포를 생성할 때 CloudFront에서는 해당하는 오리진 액세스 자격 증명을 생성합니다.

사용자 ID

> Note
>
> Amazon S3 버킷 오리진에만 적용됩니다(웹 사이트 엔드포인트로 구성된 경우는 제외).

Origin Access Identity(원본 액세스 ID)에 대해 Use an Existing Identity(기존 자격 증명 사용)를 선택한 경우, 사용하려는 원본 액세스 ID를 선택합니다. 다른 AWS 계정과 연결된 원본 액세스 ID는 사용할 수 없습니다.

버킷에 대한 읽기 권한 부여

> Note
>
> Amazon S3 버킷 오리진에만 적용됩니다(웹 사이트 엔드포인트로 구성된 경우는 제외).

CloudFront에서 원본 액세스 ID에 자동으로 Amazon S3 버킷에 있는 객체에 대한 읽기 권한을 부여하려는 경우, Yes, Update Bucket Policy(예, 버킷 정책을 업데이트합니다)를 선택합니다.

> **Important**
>
> Yes, Update Bucket Policy(예, 버킷 정책을 업데이트합니다)를 선택한 경우, CloudFront에서는 지정한 원본 액세스 ID에 버킷의 객체에 대한 읽기 권한을 부여하도록 버킷 정책을 업데이트합니다. 그러나, CloudFront에서는 버킷 정책에 있는 기존 권한이나 개별 객체에 대한 권한을 제거하지는 않습니다. 즉, 사용자가 Amazon S3 URL을 사용하는 버킷의 객체에 대해 현재 액세스 권한이 있는 경우, CloudFront에서 버킷 정책을 업데이트한 후에도 해당 권한은 유지됩니다. 기존 버킷 정책과 버킷의 객체에 대한 기존 권한을 확인 또는 변경하려면 Amazon S3에서 제공되는 메소드를 사용하십시오. 자세한 내용은 OAI에 Amazon S3 버킷 내 파일 읽기 권한 부여 (p. 179) 단원을 참조하십시오.

버킷 권한을 업데이트하는 대신 객체의 ACL을 업데이트하려는 경우와 같이 권한을 수동으로 업데이트하려는 경우, No, I will Update Permissions(아니요, 권한을 업데이트하겠습니다)를 선택합니다.

최소 오리진 SSL 프로토콜

오리진에 대한 HTTPS 연결을 설정할 때 CloudFront가 사용할 수 있는 최소 TLS/SSL 프로토콜을 선택합니다. TLS 프로토콜이 낮을수록 보안이 낮으므로, 오리진이 지원하는 최신 TLS 프로토콜을 선택하는 것이 좋습니다.

CloudFront API를 사용하여 CloudFront에서 TLS/SSL 프로토콜을 사용하도록 설정하면 최소 프로토콜을 설정할 수 없습니다. 대신, CloudFront에서 오리진과 함께 사용할 수 있는 모든 TLS/SSL 프로토콜을 지정합니다. 자세한 내용은 Amazon CloudFront API Reference의 OriginSslProtocols를 참조하십시오.

> **Note**
>
> 이 옵션은 웹 사이트 엔드포인트로 구성된 경우가 아니면 Amazon S3 버킷에 적용되지 않습니다. 오리진이 웹 사이트 엔드포인트로 구성되지 않은 Amazon S3 버킷인 경우, CloudFront는 항상 TLSv1.2를 사용합니다.

오리진 프로토콜 정책

> **Note**
>
> 이 옵션은 웹 사이트 엔드포인트로 구성된 경우가 아니면 Amazon S3 버킷에 적용되지 않습니다.

오리진 서버에서 객체를 가져올 때 CloudFront에서 사용하려는 프로토콜 정책입니다.

다음 값 중 하나를 선택합니다.

- HTTP Only(HTTP만 해당): CloudFront HTTP만 사용하여 오리진에 액세스합니다.

  > **Important**
  >
  > 오리진이 웹 사이트 엔드포인트로 구성된 Amazon S3 버킷인 경우 이 옵션을 선택해야 합니다. Amazon S3에서는 웹 사이트 엔트포인트에서 HTTPS 연결을 지원하지 않습니다.

- HTTPS Only(HTTPS만 해당): CloudFront HTTPS만 사용하여 오리진에 액세스합니다.
- Match Viewer: CloudFront가 뷰어 요청의 프로토콜에 따라 HTTP 또는 HTTPS를 사용하여 원본과 통신합니다. 뷰어가 HTTP 및 HTTPS 프로토콜 모두를 사용하여 요청하더라도 CloudFront에서는 한 번만 객체를 캐싱합니다.

  > **Important**
  >
  > CloudFront에서 이 오리진에 전달하는 HTTPS 최종 사용자 요청의 경우, 오리진 서버의 SSL 인증서에 있는 도메인 이름 중 하나가 Origin Domain Name(오리진 도메인 이름)으로 지정한 도메인 이름과 일치해야 합니다. 그렇지 않은 경우 CloudFront은(는) 최종 사용자 요청에 대해 요청

된 객체를 반환하는 대신 HTTP 상태 코드 502(잘못된 게이트웨이)로 응답합니다. 자세한 내용은 CloudFront에서 SSL/TLS 인증서를 사용하기 위한 요구 사항 (p. 104) 단원을 참조하십시오.

오리진 응답 제한 시간

Note

이 옵션은 웹 사이트 엔드포인트로 구성된 경우가 아니면 Amazon S3 버킷에 적용되지 않습니다.

오리진 응답 제한 시간(오리진 읽기 제한 시간 또는 오리진 요청 제한 시간이라고도 함)은 다음 두 값에 모두 적용됩니다.

- CloudFront이(가) 사용자 지정 오리진에 요청을 전달한 후 응답을 기다리는 시간(초)
- CloudFront이(가) 오리진으로부터 응답 패킷을 수신한 후 다음 패킷을 수신할 때까지 대기하는 시간(초)

기본 제한 시간은 30초입니다. 이 값을 4초에서 60초 사이로 변경할 수 있습니다. 해당 범위를 벗어나는 제한 시간 값이 필요한 경우 AWS 지원 센터에서 사례를 생성합니다.

Tip

최종 사용자가 HTTP 504 상태 코드 오류를 경험하고 있기 때문에 오리진 응답 제한 시간 값을 늘리고 싶은 경우에는 제한 시간 값을 변경하기 앞서 이러한 오류를 제거할 수 있는 다른 방법들을 시도해 보십시오. HTTP 504 상태 코드(게이트웨이 제한 시간) (p. 226)에서 문제 해결 주제를 참조하십시오.

CloudFront 동작은 최종 사용자 요청 내 HTTP 메서드에 따라 달라집니다.

- `GET` 및 `HEAD` 요청 – 오리진이 읽기 제한 시간이 경과할 때까지 응답하지 않거나 오리진이 구성된 제한 시간 동안 응답을 멈춘 경우, CloudFront는 연결을 끊고 오리진에 두 번 더 연결을 시도합니다. 오리진이 세 번째 시도에 응답하지 않는 경우, CloudFront는 동일한 오리진의 콘텐츠에 대해 다른 요청을 수신할 때까지 다시 시도하지 않습니다.
- `DELETE`, `OPTIONS`, `PATCH`, `PUT` 및 `POST` 요청 – 오리진이 읽기 제한 시간이 경과할 때까지 응답하지 않는 경우, CloudFront는 연결을 끊고 오리진에 다시 연결을 시도하지 않습니다. 필요한 경우 클라이언트는 요청을 다시 제출할 수 있습니다.

오리진 연결 유지 제한 시간

Note

이 옵션은 웹 사이트 엔드포인트로 구성된 경우가 아니면 Amazon S3 버킷에 적용되지 않습니다.

CloudFront이(가) 응답의 마지막 패킷을 가져온 후 연결을 유지하기 위해 시도하는 시간(초). 지속적인 연결을 유지하면 TCP 연결 재설정 및 이후 요청에 대한 별도의 TLS 핸드쉐이크 수행에 필요한 시간이 절약됩니다. 연결 유지 제한 시간을 늘리면 배포에서 연결당 요청 지표를 개선할 수 있습니다.

Note

오리진 연결 유지 제한 시간 값이 유효하려면 오리진이 지속적 연결을 허용하도록 구성되어야 합니다.

기본 제한 시간은 5초입니다. 이 값을 1초에서 60초 사이로 변경할 수 있습니다. 60초보다 긴 연결 유지 제한 시간이 필요한 경우 AWS 지원 센터에서 사례를 생성합니다.

HTTP 포트

Note

이 옵션은 웹 사이트 엔드포인트로 구성된 경우가 아니면 Amazon S3 버킷에 적용되지 않습니다.

선택 사항입니다. 사용자 지정 오리진에서 수신 대기하는 HTTP 포트입니다. 유효한 값에는 포트 80, 443 및 1024~65535가 포함됩니다. 기본값은 포트 80입니다.

HTTPS 포트

> Note
>
> 이 옵션은 웹 사이트 엔드포인트로 구성된 경우가 아니면 Amazon S3 버킷에 적용되지 않습니다.

선택 사항입니다. 사용자 지정 오리진에서 수신 대기하는 HTTPS 포트입니다. 유효한 값에는 포트 80, 443 및 1024~65535가 포함됩니다. 기본값은 포트 443입니다.

오리진 사용자 지정 헤더

CloudFront이 오리진에 요청을 전달할 때마다 사용자 지정 헤더를 포함하도록 하려면 다음 값을 지정합니다.

헤더 이름

CloudFront이 오리진에 전달하도록 하려는 헤더의 이름입니다.

값

Custom Header(사용자 지정 헤더) 필드에 지정한 헤더의 값입니다.

자세한 내용은 사용자 지정 헤더를 오리진 요청에 추가 (p. 248) 단원을 참조하십시오.

현재 오리진에 전달할 수 있는 사용자 지정 헤더의 최대 수, 사용자 지정 헤더 이름과 값의 최대 길이 및 모든 헤더 이름 및 값의 최대 총 길이는 할당량 (p. 447) 단원을 참조하십시오.

캐시 동작 설정

캐시 동작은 웹 사이트에 있는 파일의 지정된 URL 경로 패턴에 대해 다양한 CloudFront 기능을 구성할 수 있도록 합니다. 예를 들어, 하나의 캐시 동작이 CloudFront용 오리진 서버로 사용 중인 웹 서버의 `images` 디렉터리에 있는 전체 `.jpg` 파일에 적용될 수 있습니다. 각 캐시 동작에 대해 구성 가능한 기능에는 다음이 포함됩니다.

- 경로 패턴
- CloudFront 배포에 대해 다수의 오리진을 구성한 경우, CloudFront에서 요청을 전달하려는 대상 오리진
- 쿼리 문자열을 오리진에 전달할지 여부
- 지정된 파일에 액세스하기 위해 서명된 URL이 필요한지 여부
- 사용자가 HTTPS를 사용하여 그러한 파일에 액세스하도록 지정할지 여부
- 오리진에서 파일에 추가한 `Cache-Control` 헤더의 값과 관계없이 그러한 파일이 CloudFront 캐시에 머무르는 최소 시간

새 배포를 만든 경우, 배포를 만들 때 지정한 오리진에 모든 요청을 자동으로 전달하는 기본 캐시 동작에 대해 설정을 지정합니다. 배포를 만든 후에는 경로 패턴과 일치하는 객체(예: `*.jpg`)에 대한 요청을 받았을 때 CloudFront에서 응답하는 방식을 정의하는 추가 캐시 동작을 만들 수 있습니다. 추가 캐시 동작을 만든 경우 기본 캐시 동작은 항상 마지막으로 처리됩니다. 나머지 캐시 동작은 CloudFront 콘솔에 나열한 순서에 따라 처리됩니다. 또는 CloudFront API를 사용하는 경우 배포에 대한 `DistributionConfig` 요소에 나열한 순서대로 처리됩니다. 자세한 내용은 경로 패턴 (p. 44) 단원을 참조하십시오.

캐시 동작을 만들 때 CloudFront에서 객체를 가져오려는 대상 오리진을 하나만 지정합니다. CloudFront에서 오리진 전체로부터 객체를 배포하려는 경우, 기본 캐시 동작을 포함해 캐시 동작은 최소한 보유한 오리진의 수만큼 있어야 합니다. 예를 들어, 두 개의 오리진이 있는 경우 기본 캐시 동작이 하나 뿐이라면, 이 기본 캐시

동작으로 인해 CloudFront에서는 두 개의 오리진 중 하나에서 객체를 가져오지만, 나머지 오리진은 사용되지 않습니다.

현재 하나의 배포에 추가할 수 있는 캐시 동작의 최대 수를 살펴보고 더 높은 할당량(이전에는 제한이라고 함)을 요청하려면, 웹 배포의 일반 할당량 (p. 447) 단원을 참조하십시오.

경로 패턴

경로 패턴(예: `images/*.jpg`)은 이 캐시 동작을 적용하려는 요청을 지정합니다. CloudFront에서 최종 사용자 요청을 받으면, 요청된 경로는 배포에 나열된 캐시 동작의 순서대로 경로 패턴과 비교됩니다. 첫 번째 일치 결과로 해당 요청에 적용되는 캐시 동작이 결정됩니다. 예를 들어, 다음과 같은 3개의 경로 패턴과 순서로 3개의 캐시 동작이 있다고 가정합니다.

- `images/*.jpg`
- `images/*`
- `*.gif`

> **Note**
>
> `/images/*.jpg`와 같이 경로 패턴 시작 부분에 슬래시(/)를 포함하도록 선택할 수 있습니다. / 포함 여부와 상관 없이 CloudFront의 동작은 동일합니다.

`images/sample.gif` 파일에 대한 요청은 첫 번째 경로 패턴을 충족하지 않으므로 연결된 캐시 동작은 이 요청에 적용되지 않습니다. 이 파일은 두 번째 경로 패턴을 만족하므로 세 번째 경로 패턴과도 일치하더라도 두 번째 경로 패턴과 연결된 캐시 동작이 이 요청에 적용됩니다.

> **Note**
>
> 새 배포를 만들 때 기본 캐시 동작에 대한 경로 패턴의 값은 *(모든 파일)로 설정되고 이는 변경할 수 없습니다. 이 값에 따라 CloudFront에서는 객체에 대한 모든 요청을 Origin Domain Name (p. 38) 필드에서 지정한 오리진으로 전달합니다. 객체에 대한 요청이 나머지 캐시 동작 중 어느 것의 경로 패턴과도 일치하지 않는 경우, CloudFront에서는 기본 캐시 동작에서 지정한 동작을 적용합니다.
>
> **Important**
>
> 경로 패턴과 그 순서를 주의 깊게 정의하지 않으면 사용자에게 콘텐츠에 대한 원치 않는 액세스 권한을 주게 될 수 있습니다. 예를 들어, 하나의 요청이 두 캐시 동작의 경로 패턴과 일치한다고 가정합니다. 첫 번째 캐시 동작은 서명된 URL이 필요하고 두 번째 캐시 동작은 서명된 URL이 필요하지 않습니다. CloudFront에서는 첫 번째 일치 결과와 연결된 캐시 동작을 처리하므로 사용자는 서명된 URL을 사용하지 않고도 객체에 액세스할 수 있습니다.

MediaPackage 채널을 사용하는 경우, 오리진의 엔드포인트 유형에 대해 정의하는 캐시 동작을 위한 특정 경로 패턴을 포함시켜야 합니다. 예를 들어 DASH 엔드포인트의 경우 경로 패턴에 `*.mpd`를 입력합니다. 자세한 정보와 특정 지침은 AWS Elemental MediaPackage를 사용하여 포맷된 라이브 비디오 제공 (p. 265) 단원을 참조하십시오.

사용자가 지정한 경로는 지정된 디렉터리와 이 디렉터리의 하위 디렉터리에 있는 모든 파일에 대한 요청에 적용됩니다. CloudFront에서는 경로 패턴을 평가할 때 쿼리 문자열 또는 쿠키를 고려하지 않습니다. 예를 들어, `images` 디렉터리에 `product1` 및 `product2` 하위 디렉터리가 포함되어 있으면, 경로 패턴 `images/*.jpg`는 모든 `images`, `images/product1`, `images/product2` 디렉터리의 모든 .jpg 파일에 대한 요청에 적용됩니다. 다른 캐시 동작을 `images/product1` 및 `images` 디렉터리의 파일이 아닌 `images/product2` 디렉터리의 파일에 적용하려는 경우, `images/product1`에 대한 별도의 캐시 동작을 만들어 `images` 디렉터리에 대한 캐시 동작보다 위로(이전으로) 해당 캐시 동작을 이동합니다.

다음 와일드카드 문자를 경로 패턴에 사용할 수 있습니다.

- `*`는 0개 이상의 문자에 해당합니다.

- ?는 정확히 1문자에 해당합니다.

다음 예는 와일드카드 문자가 어떻게 작동하는지 보여줍니다.

경로 패턴	경로 패턴과 일치하는 파일
`*.jpg`	모든 .jpg 파일
`images/*.jpg`	`images` 디렉터리와 `images` 디렉터리의 하위 디렉터리에 있는 모든 .jpg 파일
`a*.jpg`	• a로 시작하는 파일 이름을 지닌 모든 .jpg 파일(예: `apple.jpg`, `appalachian_trail_2012_05_21.jpg`) • a로 시작하는 파일 경로에 있는 모든 .jpg 파일(예: `abra/cadabra/magic.jpg`)
`a??.jpg`	a로 시작하여 나머지는 정확히 두 문자가 오는 파일 이름을 지닌 모든 .jpg 파일(예: `ant.jpg`, `abe.jpg`)
`*.doc*`	파일 이름 확장명이 `.doc`로 시작하는 모든 .jpg 파일(예: `.doc`, `.docx`, `.docm` 파일). 이 경우, `*.doc?` 경로 패턴을 사용할 수 없습니다. 이 경로 패턴은 `.doc` 파일에 대한 요청에 적용할 수 없기 때문입니다. ? 와일드카드 문자는 정확히 한 문자를 대체합니다.

경로 패턴의 최대 길이는 255자입니다. 이 값에는 다음 문자 중 어느 것이든 포함할 수 있습니다.

- A~Z, a~z

  경로 패턴은 대소문자를 구분하므로 경로 패턴 `*.jpg`는 `LOGO.JPG` 파일에 적용되지 않습니다.
- 0~9
- _ - . * $ / ~ " ' @ : +
- &, `&`로 처리되고 반환됨

Origin(기존 배포에만 해당)

기존 오리진에 대한 Origin ID(오리진 ID)의 값을 입력합니다. 이를 통해 요청(예: http://example.com/logo.jpg)이 캐시 동작의 경로 패턴(예: *.jpg)이나 기본 캐시 동작(*)과 일치할 때 CloudFront에서 요청을 라우팅하려는 오리진을 식별합니다.

Viewer Protocol Policy

최종 사용자가 CloudFront 엣지 로케이션의 콘텐츠를 액세스하는 데 사용하게 하려는 프로토콜 정책을 선택합니다.

- HTTP 및 HTTPS: 최종 사용자는 두 프로토콜 모두를 사용할 수 있습니다.
- Redirect HTTP to HTTPS(HTTP를 HTTPS로 재지정): 최종 사용자는 두 프로토콜 모두 사용할 수 있지만, HTTP 요청은 자동으로 HTTPS 요청으로 리디렉션됩니다.
- HTTPS Only(HTTPS만 해당): 최종 사용자는 HTTPS를 사용하는 경우에만 콘텐츠에 액세스할 수 있습니다.

자세한 내용은 최종 사용자와 CloudFront 간의 통신에 HTTPS 요구 (p. 93) 단원을 참조하십시오.

필드 레벨 암호화

특정 데이터 필드에 필드 레벨 암호화를 적용하고자 하는 경우 드롭다운 목록에서 필드 레벨 암호화 구성을 선택합니다.

자세한 내용은 필드 레벨 암호화 사용으로 민감한 데이터 보호 (p. 185) 단원을 참조하십시오.

Allowed HTTP Methods

CloudFront에서 오리진을 처리하고 전달하기 위한 HTTP 메소드를 지정합니다.

- GET, HEAD: 오리진에서 객체를 가져오거나 객체 헤더를 가져오기 위해서만 CloudFront를 사용할 수 있습니다.
- GET, HEAD, OPTIONS: 오리진에서 객체를 가져오거나 객체 헤더를 가져오기 위해, 또는 오리진 서버에서 지원되는 옵션 목록을 가져오기 위해서만 CloudFront를 사용할 수 있습니다.
- GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE: CloudFront를 사용하여 객체를 추가, 업데이트, 삭제하고 가져올 수 있으며 객체 헤더를 가져올 수 있습니다. 또한 웹 양식에서 데이터를 제출하는 등의 기타 POST 작업을 수행할 수 있습니다.

  Note

  CloudFront에서는 `GET` 및 `HEAD` 요청에 대한 응답을 캐싱하고 선택적으로 `OPTIONS` 요청에 대한 응답을 캐싱합니다. CloudFront에서는 다른 메서드를 사용하는 요청에 대한 응답을 캐싱하지 않습니다.

Amazon S3 버킷을 배포용 오리진으로 사용하는 경우와 CloudFront 원본 액세스 ID를 사용하는 경우, `POST` 요청은 일부 Amazon S3 리전에서 지원되지 않으며 해당 리전의 `PUT` 요청에는 추가 헤더가 필요합니다. 자세한 내용은 서명 버전 4 인증만 지원하는 Amazon S3 리전에서 OAI 사용 (p. 182) 단원을 참조하십시오.

Important

GET, HEAD, OPTIONS 또는 GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE를 선택하는 경우 사용자가 수행하길 원치 않는 작업을 수행하지 못하도록 Amazon S3 버킷 또는 사용자 지정 오리진에 대한 액세스를 제한해야 할 수 있습니다. 다음 예에서는 액세스를 제한하는 방법을 설명합니다.

- 배포에 대한 오리진으로 Amazon S3를 사용하려는 경우: CloudFront 원본 액세스 ID를 만들어 Amazon S3 콘텐츠에 대한 액세스를 제한하고 이 원본 액세스 ID에 적절한 사용 권한을 부여해야 합니다. 예를 들어, `PUT`을 사용할 의도로 이러한 메서드만 허용 및 전달하도록 CloudFront를 구성한 경우, `DELETE` 요청을 적절히 처리하도록 Amazon S3 버킷 정책 또는 ACL을 구성해야 합니다. 자세한 내용은 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오.
- 사용자 지정 오리진을 사용하는 경우: 모든 메서드를 처리하도록 오리진 서버를 구성합니다. 예를 들어, `POST`를 사용할 의도로 이러한 메서드만 허용 및 전달하도록 CloudFront를 구성한 경우, `DELETE` 요청을 적절히 처리하도록 오리진 서버를 구성해야 합니다.

Cached HTTP Methods

최종 사용자가 `OPTIONS` 요청을 제출할 때 CloudFront에서 오리진의 응답을 캐싱할지 여부를 지정합니다. CloudFront에서는 항상 `GET` 및 `HEAD` 요청에 대한 응답을 캐싱합니다.

선택한 요청 헤더 기반의 캐시

CloudFront에서 지정된 헤더의 값을 기반으로 객체를 캐싱할지 여부를 지정합니다.

- None (improves caching)(없음(캐싱 향상)) – CloudFront가 헤더 값을 기반으로 객체를 캐시하지 않습니다.
- Whitelist(화이트리스트) – CloudFront가 지정된 헤더의 값만을 기반으로 객체를 캐시합니다. Whitelist Headers(화이트리스트 헤더)를 사용하여 CloudFront에서 캐시의 기준으로 설정할 헤더를 선택합니다.
- 모두 – CloudFront가 이 캐시 동작과 연결된 객체를 캐싱하지 않습니다. 그 대신, CloudFront가 모든 요청을 오리진에 전송합니다. (Amazon S3 오리진에는 권장하지 않음)

어떤 옵션을 선택하든 상관없이, CloudFront는 오리진에 특정 헤더를 전달하고 전달한 헤더에 따라 작업을 수행합니다. CloudFront가 헤더 전달을 처리하는 방법에 대한 자세한 내용은 HTTP 요청 헤더 및 CloudFront 동작(사용자 지정 및 S3 오리진) (p. 239) 단원을 참조하십시오.

요청 헤더를 사용하여 CloudFront에서 캐싱을 구성하는 방법에 대한 자세한 내용은 요청 헤더에 따라 콘텐츠 캐싱 (p. 200) 단원을 참조하십시오.

Whitelist Headers

객체를 캐싱할 때 CloudFront에서 고려할 헤더를 지정합니다. 사용 가능한 헤더 목록에서 헤더를 선택하고 추가를 선택합니다. 사용자 지정 헤더를 전달하려면 필드에 헤더 이름을 입력하고 Add Custom(사용자 지정 항목 추가)을 선택합니다.

현재 각 캐시 동작에 대해 화이트리스트로 지정할 수 있는 헤더의 최대 수를 살펴보고 더 높은 할당량(이전에는 제한이라고 함)을 요청하려면, 사용자 지정 헤더에 대한 할당량(웹 배포만 해당) (p. 449) 단원을 참조하십시오.

Object Caching

오리진 서버에서 객체에 `Cache-Control` 헤더를 추가하여 CloudFront 캐시에 객체가 머무르는 시간을 제어하는 경우와, `Cache-Control` 값을 변경하지 않으려는 경우 Use Origin Cache Headers(오리진 캐시 헤더 사용)를 선택합니다.

`Cache-Control` 헤더와 관계없이 CloudFront 캐시에 객체가 머무르는 최소 및 최대 시간과 객체에서 `Cache-Control` 헤더가 누락된 경우 CloudFront 캐시에 객체가 머무르는 기본 시간을 지정하려면 Customize(사용자 지정)를 선택합니다. 그런 다음 최소 TTL, 기본 TTL 및 최대 TTL 필드에서 값을 지정합니다.

자세한 내용은 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204) 단원을 참조하십시오.

Minimum TTL

객체가 업데이트되었는지 여부를 결정하도록 CloudFront에서 오리진에 다른 요청을 전달하기 전에 CloudFront 캐시에서 객체를 머무르게 하려는 최소 시간을 초 단위로 지정합니다. 최소 TTL의 기본값은 0초입니다.

> **Important**
>
> 하나의 캐시 동작에 대해 전체 헤더를 오리진에 전달하도록 CloudFront를 구성한 경우, CloudFront에서는 연결된 객체를 캐시하지 않습니다. 대신 CloudFront에서는 그러한 객체에 대한 전체 요청을 오리진에 전달합니다. 그 구성에서 최소 TTL의 값은 0이어야 합니다.

최소 TTL의 값을 지정하려면 Object Caching(객체 캐싱) 설정의 Customize(사용자 지정) 옵션을 선택해야 합니다.

자세한 내용은 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204) 단원을 참조하십시오.

Maximum TTL

객체가 업데이트되었는지 여부를 확인하도록 CloudFront에서 오리진에 쿼리하기 전에 CloudFront 캐시에서 객체를 머무르게 하려는 최대 시간을 초 단위로 지정합니다. 최대 TTL에 지정하는 값은 오리진이 객체에 `Cache-Control max-age`, `Cache-Control s-maxage` 또는 `Expires` 등의 HTTP 헤더를 추가할 경우에만 적용됩니다. 자세한 내용은 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204) 단원을 참조하십시오.

최대 TTL의 값을 지정하려면 Object Caching(객체 캐싱) 설정의 Customize(사용자 지정) 옵션을 선택해야 합니다.

최대 TTL의 기본값은 31536000초(1년)입니다. 최소 TTL 또는 기본 TTL의 값을 31536000초 이상으로 변경할 경우 최대 TTL의 기본값은 기본 TTL의 값으로 변경됩니다.

기본 TTL

객체가 업데이트되었는지 여부를 결정하도록 CloudFront에서 오리진에 다른 요청을 전달하기 전에 CloudFront 캐시에서 객체를 머무르게 하려는 기본 시간을 초 단위로 지정합니다. 기본 TTL에 지정하는 값은 오리진이 객체에 `Cache-Control max-age`, `Cache-Control s-maxage` 또는 `Expires` 등의 HTTP 헤더를 추가하지 않을 경우에만 적용됩니다. 자세한 내용은 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204) 단원을 참조하십시오.

기본 TTL에 값을 지정하려면 Object Caching(객체 캐싱) 설정에 대해 Customize(사용자 지정) 옵션을 선택해야 합니다.

기본 TTL의 기본값은 86400초(1일)입니다. Minimum TTL(최소 TTL)의 값을 86,400초 이상으로 변경할 경우 Default TTL(기본 TTL)의 기본값은 Minimum TTL(최소 TTL)의 값으로 변경됩니다.

쿠키 전달

> **Note**
>
> 이 옵션은 웹 사이트 엔드포인트로 구성된 경우가 아니면 Amazon S3 버킷에 적용되지 않습니다.

CloudFront에서 쿠키를 오리진 서버로 전달할지 여부와 전달할 쿠키(전달하는 경우)를 지정합니다. 선택한 쿠키(쿠키의 화이트리스트)만 전달하도록 선택한 경우, 쿠키 이름을 Whitelist Cookies(화이트리스트 쿠키) 필드에 입력합니다. 모두를 선택하는 경우 CloudFront에서는 애플리케이션에서 사용하는 쿠키 수와 관계없이 모든 쿠키를 전달합니다.

Amazon S3에서는 쿠키를 처리하지 않으며 오리진에 쿠키를 전달함으로써 캐싱 가능성이 줄어듭니다. Amazon S3 오리진에 요청을 전달하는 캐시 동작의 경우 Forward Cookies(쿠키 전달)에 대해 없음을 선택합니다.

오리진으로 쿠키를 전달하는 것에 대한 자세한 내용은 쿠키를 기반으로 콘텐츠 캐싱 (p. 198) 단원을 참조하십시오.

화이트리스트 쿠키

> **Note**
>
> 이 옵션은 웹 사이트 엔드포인트로 구성된 경우가 아니면 Amazon S3 버킷에 적용되지 않습니다.

Forward Cookies(쿠키 전달) 목록의 Whitelist(화이트리스트)를 선택하는 경우, Whitelist Cookies(화이트리스트 쿠키) 필드에 이 캐시 동작에 대해 CloudFront에서 오리진 서버에 전달하려는 쿠키 이름을 입력합니다. 각 쿠키 이름을 새 줄에 입력합니다.

다음 와일드카드를 지정하여 쿠키 이름을 지정할 수 있습니다.

- *는 쿠키 이름의 0개 이상의 문자에 해당합니다.
- ?는 쿠키 이름의 정확히 1문자에 해당합니다.

예를 들어, 다음과 같은 쿠키 이름이 포함된 객체에 대한 최종 사용자 요청을 가정합니다.

`userid_`*`member-number`*

각각의 사용자는 *`member-number`*에 대한 고유한 값을 갖습니다. 이 경우에는 CloudFront에서 각 멤버별로 개별 버전의 객체를 캐싱하려 합니다. 모든 쿠키를 오리진에 전달함으로써 이를 달성할 수는 있지만, 최종 사용자 요청에 CloudFront에서 캐싱하길 원치 않는 쿠키가 일부 포함됩니다. 또는, 다음 값을 쿠키 이름으로 지

정할 수 있습니다. 이렇게 하면 CloudFront에서 해당 오리진에 `userid_`로 시작하는 모든 쿠키를 전달합니다.

`userid_*`

현재 각 캐시 동작에 대해 화이트리스트로 지정할 수 있는 쿠키 이름의 최대 수를 살펴보고 더 높은 할당량(이전에는 제한이라고 함)을 요청하려면, 화이트리스트에 등록된 쿠키에 대한 할당량(웹 배포만 해당) (p. 448) 단원을 참조하십시오.

Query String Forwarding and Caching

CloudFront는 쿼리 문자열 파라미터 값을 기준으로 콘텐츠의 다른 버전을 캐싱할 수 있습니다. 다음 옵션 중 하나를 선택합니다.

None (Improves Caching)

오리진에서 쿼리 문자열 파라미터 값과 상관 없이 객체의 동일한 버전을 반환하는 경우 이 옵션을 선택합니다. 이렇게 하면 CloudFront에서 캐시로부터 요청을 제공할 수 있는 가능성을 높이고, 그에 따라 성능이 향상되고 오리진에 걸리는 부하를 줄여 줍니다.

Forward all, cache based on whitelist

오리진 서버에서 하나 이상의 쿼리 문자열 파라미터를 기준으로 다른 버전의 객체를 반환할 경우 이 옵션을 선택합니다. 그런 다음 Query String Whitelist (p. 49) 필드에서 CloudFront가 이 캐싱의 기준으로 사용하도록 하려는 파라미터를 지정합니다.

Forward all, cache based on all

오리진 서버에서 모든 쿼리 문자열 파라미터를 기준으로 다른 버전의 객체를 반환할 경우 이 옵션을 선택합니다.

성능 향상을 포함하여 쿼리 문자열 파라미터를 기준으로 하는 캐싱에 대한 자세한 내용은 쿼리 문자열 파라미터 기반의 콘텐츠 캐싱 (p. 196) 단원을 참조하십시오.

Query String Whitelist

Query String Forwarding and Caching (p. 49)에서 Forward all, cache based on whitelist(모두 전달, 화이트리스트 기반으로 캐싱)를 선택하면 CloudFront에서 캐싱의 기준으로 사용하도록 하려는 쿼리 문자열 파라미터를 지정합니다.

Smooth Streaming

Microsoft Smooth Streaming 포맷의 미디어 파일을 배포하려는데 IIS 서버가 없으면 예를 선택합니다.

Microsoft Smooth Streaming 포맷의 미디어 파일을 배포하기 위한 오리진으로 사용하려는 Microsoft IIS 서버가 있거나, Smooth Streaming 미디어 파일을 배포하지 않으려는 경우 아니요를 선택합니다.

> **Note**
>
> 예를 지정한 경우에도 콘텐츠가 경로 패턴 값과 일치하는 경우 이 캐시 동작을 사용하여 다른 콘텐츠를 배포할 수 있습니다.

자세한 내용은 Microsoft Smooth Streaming용 온디맨드 비디오 구성 (p. 262) 단원을 참조하십시오.

Restrict Viewer Access (Use Signed URLs)

이 캐시 동작의 `PathPattern`과 일치하는 객체에 대한 요청에 퍼블릭 URL을 사용하려는 경우 아니요를 선택합니다.

이 캐시 동작의 `PathPattern`과 일치하는 객체에 대한 요청에 서명된 URL을 사용하려는 경우 예를 선택합니다. 그런 다음, 서명된 URL 생성에 사용하려는 AWS 계정을 지정합니다. 이러한 계정은 신뢰할 수 있는 서명자라고도 합니다.

신뢰할 수 있는 서명자에 대한 자세한 내용은 서명된 URL 및 서명된 쿠키(신뢰할 수 있는 서명자)를 만들 수 있는 AWS 계정 지정 (p. 118) 단원을 참조하십시오.

Trusted Signers

이 캐시 동작에 대해 신뢰할 수 있는 서명자로 사용하려는 AWS 계정을 선택합니다.

- 셀프: 현재 AWS Management 콘솔에 신뢰할 수 있는 서명자로 로그인된 계정을 사용합니다. 현재 IAM 사용자로 로그인되어 있는 경우, 연결된 AWS 계정이 신뢰할 수 있는 서명자로 추가됩니다.
- Specify Accounts(계정 지정): 신뢰할 수 있는 서명자에 대한 계정 번호를 AWS Account Numbers(AWS 계정 번호) 필드에 입력합니다.

서명된 URL을 생성하려면 AWS 계정에 하나 이상의 활성 CloudFront 키 페어가 있어야 합니다.

> **Important**
>
> 콘텐츠 배포에 이미 사용 중인 배포를 업데이트하려는 경우, 서명된 URL 생성을 시작할 준비가 되었을 때만 신뢰할 수 있는 서명자를 추가합니다. 배포에 신뢰할 수 있는 서명자를 추가한 후에, 사용자는 서명된 URL을 사용하여 이 캐시 동작의 `PathPattern`과 일치하는 객체에 액세스해야 합니다.

AWS Account Numbers

현재 계정에 추가로 또는 현재 계정 대신에 AWS 계정을 사용하여 서명된 URL을 생성하려는 경우, 이 필드에 한 줄당 AWS 계정 번호 하나씩을 입력합니다. 다음을 참조하십시오.

- 지정하는 계정에는 하나 이상의 활성 CloudFront 키 페어가 있어야 합니다. 자세한 내용은 신뢰할 수 있는 서명자에 대해 CloudFront 키 페어 생성 (p. 119) 단원을 참조하십시오.
- IAM 사용자에 대해 CloudFront 키 페어를 생성할 수 없으므로 IAM 사용자는 신뢰할 수 있는 서명자로 사용할 수 없습니다.
- 계정에 대해 AWS 계정 번호를 가져오는 방법에 대한 자세한 내용은 Amazon Web Services 일반 참조의 보안 자격 증명을 가져오는 방법 단원을 참조하십시오.
- 현재 계정에 대한 계정 번호를 입력하면 CloudFront에서는 셀프 확인란이 자동으로 선택되고 AWS Account Numbers(AWS 계정 번호) 목록에서 계정 번호가 삭제됩니다.

자동으로 객체를 압축

CloudFront가 최종 사용자 요청이 요청 헤더에 `Accept-Encoding: gzip`을 포함하는 경우 특정 유형의 파일을 자동 압축하려면 예를 선택합니다. CloudFront가 콘텐츠를 압축하면 파일 크기가 더 작아지므로 다운로드 속도가 빨라지고 웹 페이지는 사용자에게 더 빨리 렌더링됩니다. 자세한 내용은 압축된 파일 제공 (p. 87) 단원을 참조하십시오.

이벤트 유형

다음 CloudFront 이벤트가 하나 이상 발생할 때 Lambda 함수를 실행하도록 선택할 수 있습니다.

- CloudFront가 최종 사용자의 요청을 수신할 때(최종 사용자 요청)
- CloudFront가 오리진에 요청을 전달하기 전(오리진 요청)
- CloudFront가 오리진의 응답을 수신할 때(오리진 응답)
- CloudFront가 최종 사용자에게 응답을 반환하기 전(최종 사용자 응답)

자세한 내용은 Lambda 함수를 트리거하는 데 사용할 CloudFront 이벤트를 결정하는 방법 (p. 304) 단원을 참조하십시오.

Lambda함수 ARN

트리거를 추가할 Lambda 함수의 Amazon 리소스 이름(ARN)을 지정합니다. 함수의 ARN을 가져오는 방법에 대한 자세한 내용은 CloudFront 콘솔을 사용한 트리거 추가 절차의 1단계를 참조하십시오.

배포 세부 정보

다음 값들이 전체 배포에 적용됩니다.

요금 계층

CloudFront 서비스를 위해 지불하려는 최고 가격에 해당하는 가격 등급을 선택합니다. 기본적으로 CloudFront는 모든 CloudFront 지역의 에지에서 객체를 제공합니다.

가격 등급과 가격 등급 선택이 배포를 위한 CloudFront 성능에 미치는 영향에 대한 자세한 내용은 CloudFront 배포의 가격 등급 선택 (p. 9) 단원을 참조하십시오. 가격 등급이 CloudFront 지역에 매핑되는 방식을 포함해 CloudFront 요금에 대한 자세한 내용은 Amazon CloudFront 요금을 참조하십시오.

AWS WAF 웹 ACL

AWS WAF를 사용하여 지정하는 기준을 기반으로 요청을 허용 또는 차단하려는 경우, 이 배포와 연결할 웹 ACL을 선택합니다.

AWS WAF는 CloudFront에 전달되는 HTTP 및 HTTPS 요청을 모니터링할 수 있게 해주고 콘텐츠에 대한 액세스를 제어할 수 있게 해주는 웹 애플리케이션 방화벽입니다. 요청이 허용되는 IP 주소나 쿼리 문자열의 값으로부터 지정하는 조건 등 지정하는 조건에 따라, CloudFront는 요청된 콘텐츠나 HTTP 403 상태 코드(금지됨)로 요청에 응답합니다. 또한 요청이 차단될 때 사용자 지정 오류 페이지를 반환하도록 CloudFront를 구성할 수 있습니다. AWS WAF에 대한 자세한 내용은 AWS WAF 개발자 안내서를 참조하십시오.

Alternate Domain Names (CNAMEs)

선택. 사용자가 배포를 만들 때 CloudFront에서 할당하는 도메인 이름 대신 객체의 URL에 사용할 하나 이상의 도메인 이름을 지정합니다. 도메인 이름을 소유하거나, 도메인 이름을 사용할 권한이 있어야 합니다. 사용 권한은 SSL/TLS 인증서를 추가하여 확인합니다.

예를 들어 객체의 URL이 아래와 같은 경우

`/images/image.jpg`

아래와 같이 변경하고

`http://www.example.com/images/image.jpg`

아래와 같이 변경하지 않으려면

`http://d111111abcdef8.cloudfront.net/images/image.jpg`

`www.example.com`에 대한 CNAME을 추가합니다.

> **Important**
>
> 배포에 `www.example.com`에 대한 CNAME을 추가하는 경우, 다음도 수행해야 합니다.
>
> - DNS 서비스로 CNAME 레코드를 만들거나 업데이트해 `www.example.com`에 대한 쿼리를 `d111111abcdef8.cloudfront.net`로 라우팅합니다.

- 배포에 추가하는 도메인 이름(CNAME)이 포함된 신뢰할 수 있는 인증 기관(CA)의 인증서를 CloudFront에 추가하여 도메인 이름을 사용할 권한을 검증합니다.

  도메인에 대한 DNS 서비스 공급자를 통해 CNAME 레코드를 만들려면 권한이 있어야 합니다. 일반적으로 이 경우 도메인을 소유하고 있거나, 도메인 소유자를 위해 애플리케이션을 개발하는 중일 수도 있습니다.

현재 하나의 배포에 추가할 수 있는 대체 도메인 이름의 최대 수를 살펴보고 더 높은 할당량(이전에는 제한이라고 함)을 요청하려면, 웹 배포의 일반 할당량 (p. 447) 단원을 참조하십시오.

대체 도메인 이름에 대한 자세한 내용은 대체 도메인 이름(CNAME)을 추가하여 파일에 대해 사용자 지정 URL 사용 (p. 66) 단원을 참조하십시오. CloudFront URL에 대한 자세한 내용은 CloudFront에서 파일에 대한 URL 형식 사용자 지정 (p. 79) 단원을 참조하십시오.

SSL 인증서

배포에 사용할 대체 도메인 이름을 지정한 경우, Custom SSL Certificate(사용자 지정 SSL 인증서)를 선택한 후 대체 도메인 이름을 사용할 권한을 검증하고, 대체 도메인 이름이 포함된 인증서를 선택합니다. 최종 사용자가 HTTPS를 사용하여 객체에 액세스하게 하려는 경우, 이를 지원하는 설정을 선택합니다.

> **Note**
>
> 사용자 지정 SSL 인증서를 지정하려면 유효한 대체 도메인 이름을 지정해야 합니다. 자세한 내용은 대체 도메인 이름 사용과 관련된 요구 사항 (p. 73) 및 대체 도메인 이름과 HTTPS 사용 (p. 102) 단원을 참조하십시오.

- 기본 CloudFront 인증서(*.cloudfront.net) – CloudFront 도메인 이름을 객체(예: `https://d111111abcdef8.cloudfront.net/image1.jpg`)의 URL에 사용하려는 경우, 이 옵션을 선택합니다.
- 사용자 지정 SSL 인증서 – 자체 도메인 이름을 객체의 URL에 대체 도메인 이름(예: `https://example.com/image1.jpg`)으로 사용하려는 경우, 이 옵션을 선택합니다. 그런 다음 대체 도메인 이름이 포함된 사용할 인증서를 선택합니다. 인증서 목록에는 다음이 포함될 수 있습니다.
  - AWS Certificate Manager에서 제공하는 인증서
  - 타사 인증 기관으로부터 구매했고 ACM에 업로드한 인증서
  - 타사 인증 기관으로부터 구매했고 IAM 인증서 스토어에 업로드한 인증서

  이 설정을 선택한 경우 대체 도메인 이름만 객체 URL에 사용하는 것이 좋습니다(https://example.com/logo.jpg). CloudFront 배포 도메인 이름(https://d111111abcdef8.cloudfront.net/logo.jpg)을 사용하고 클라이언트가 SNI를 지원하지 않는 이전 뷰어를 사용하는 경우, 뷰어가 응답하는 방식은 Clients Supported(지원되는 클라이언트)에서 선택하는 값에 따라 달라집니다.
  - All Clients(모든 클라이언트): CloudFront 도메인 이름이 SSL/TLS 인증서의 도메인 이름과 일치하지 않으므로 뷰어에 경고가 표시됩니다.
  - Only Clients that Support Server Name Indication (SNI)(서버 이름 표시(SNI)를 지원하는 클라이언트만): CloudFront에서 객체를 반환하지 않고 해당 뷰어와의 연결을 끊습니다.

Clients Supported

하나 이상의 대체 도메인 이름을 지정하고 IAM 인증서 스토어에서 SSL 인증서를 하나 지정한 경우, 전체 클라이언트에 대해 적용되는 방법 또는 대부분의 클라이언트에 대해 적용되는 방법 중에 CloudFront에서 HTTPS 요청을 제공하려는 방법을 선택합니다.

- All Clients(모든 클라이언트): 모든 클라이언트에서 콘텐츠에 액세스할 수 있습니다. 그러나, 이 기능을 사용할 권한을 요청해야 하며, 월별 요금이 추가로 발생합니다.
- Only Clients that Support Server Name Indication (SNI)(서버 이름 표시(SNI)를 지원하는 클라이언트만): SNI를 지원하는 모든 최신 브라우저에서 콘텐츠에 액세스할 수 있습니다. 그러나 아직까지 이용 중인 일부

브라우저에서는 SNI가 지원되지 않습니다. 이러한 브라우저를 이용 중인 사용자는 오리진에서 직접 객체를 가져오는 등의 다른 방법을 사용하여 콘텐츠에 액세스해야 합니다.

자세한 내용은 대체 도메인 이름과 HTTPS 사용 (p. 102) 단원을 참조하십시오.

보안 정책

CloudFront가 최종 사용자(클라이언트)와의 HTTPS 연결에 사용할 보안 정책을 지정합니다. 보안 정책은 다음 두 가지 설정을 결정합니다.

- CloudFront가 최종 사용자와 통신하는 데 사용하는 최소 SSL/TLS 프로토콜
- CloudFront가 최종 사용자로 반환하는 콘텐츠를 암호화할 때 사용할 수 있는 암호입니다.

각 보안 정책에 포함되는 프로토콜과 암호를 포함해 보안 정책에 관한 자세한 내용은 최종 사용자와 CloudFront 간에 지원되는 프로토콜 및 암호 (p. 99)을 참조하십시오.

사용 가능한 보안 정책은 SSL 인증서 및 사용자 지정 SSL 클라이언트 지원(CloudFront API의 `CloudFrontDefaultCertificate` 및 `SSLSupportMethod`라고도 함)에 지정한 값에 따라 다릅니다.

- SSL 인증서가 기본 CloudFront 인증서(*.cloudfront.net)인 경우(`CloudFrontDefaultCertificate`가 API에서 `true`일 때)(보안 정책)의 CloudFront가 보안 정책을 자동으로 TLSv1로 설정합니다.
- SSL 인증서가 사용자 지정 SSL 인증서(example.com)이고 사용자 지정 SSL 클라이언트 지원이 서버 이름 표시(SNI)를 지원하는 클라이언트(권장)일 때(API에서 `CloudFrontDefaultCertificate`이 `false`이고 `SSLSupportMethod`가 `sni-only`일 때) 다음 보안 정책 중에서 선택할 수 있습니다.
  - TLSv1
  - TLSv1_2016
  - TLSv1.1_2016
  - TLSv1.2_2018

  최종 사용자가 TLSv1.2를 지원하지 않는 브라우저 또는 디바이스를 사용하지 않을 경우 TLSv1.2_2018을 선택하는 것이 좋습니다.
- SSL 인증서가 사용자 지정 SSL 인증서 (example.com) 이고 사용자 지정 SSL 클라이언트 지원이 레거시 클라이언트 지원일 때(API에서 `CloudFrontDefaultCertificatefalse`이 이고 `SSLSupportMethod`가 `vip`일 때) 다음 보안 정책 중에서 선택할 수 있습니다.
  - TLSv1
  - SSLv3

  TLSv1을 선택하는 것이 좋습니다. 이 구성에서는 TLSv1_2016, TLSv1.1_2016 및 TLSv1.2_2018 보안 정책을 CloudFront 콘솔 또는 API에서 사용할 수 없습니다. 이러한 보안 정책 중 하나를 사용할 경우 다음과 같은 옵션이 있습니다.
  - 배포에 전용 IP 주소로 레거시 클라이언트 지원이 필요한지 여부를 평가합니다. 최종 사용자가 SNI(서버 이름 표시)를 지원하는 경우에는 배포의 사용자 지정 SSL 클라이언트 지원 설정을 서버 이름 표시(SNI)를 지원하는 클라이언트로 설정하는 것이 좋습니다(API에서는 `SSLSupportMethod`를 `sni-only`로 설정). 이를 통해 사용 가능한 TLS 보안 정책을 사용할 수 있으며 CloudFront 요금도 줄일 수 있습니다.
  - 레거시 클라이언트 지원을 전용 IP 주소로 유지해야 하는 경우에는 AWS 지원 센터에서 사례를 생성하여 다른 TLS 보안 정책(TLSv1_2016, TLSv1.1_2016 또는 TLSv1.2_2018) 중 하나를 요청할 수 있습니다

    Note

    AWS 지원 센터로 연락해 이러한 변경을 요청하기 전에 다음 사항을 고려하십시오.

    - 이러한 보안 정책 중 하나(TLSv1_2016, TLSv1.1_2016 또는 TLSv1.2_2018)를 레거시 클라이언트 지원 배포에 추가하면 AWS 계정의 모든 레거시 클라이언트 지원 배포에 대한 SNI 이외의 모든 최종 사용자 요청에 보안 정책이 적용됩니다. 그러나 최종 사용자가 레거시 클라이언트 지원을 사용하여 배포에 SNI 요청을 보낼 경우 해당 배포의 보안 정책이 적용됩니다. 원

하는 보안 정책이 AWS 계정의 모든 레거시 클라이언트 지원 배포에 전송된 모든 최종 사용자 요청에 적용되도록 하려면 원하는 보안 정책을 각 배포에 개별적으로 추가합니다.

- 당연히 새 보안 정책에서는 이전 보안 정책과 동일한 암호 및 프로토콜을 지원하지 않습니다. 예를 들어, 배포의 보안 정책을 TLSv1에서 TLSv1.1_2016으로 업그레이드하도록 선택한 경우 해당 배포는 DES-CBC3-SHA 암호를 더 이상 지원하지 않습니다. 각 보안 정책이 지원하는 암호 및 프로토콜에 대한 자세한 내용은 최종 사용자와 CloudFront 간에 지원되는 프로토콜 및 암호 (p. 99) 단원을 참조하십시오.

최소 SSL/TLS 프로토콜 버전

보안 정책 (p. 53) 단원을 참조하십시오.

Supported HTTP Versions

최종 사용자가 CloudFront와 통신하는 데 사용하도록 하려는 HTTP 버전을 선택합니다. 최종 사용자는 CloudFront에서 사용하도록 구성한 최신 버전을 사용합니다. HTTP/2를 지원하지 않는 최종 사용자는 자동으로 이전 버전을 사용하게 됩니다.

최종 사용자와 CloudFront가 HTTP/2를 사용하는 경우 최종 사용자는 TLS 1.2 이상을 지원해야 하며 SNI(Server Name Identification)를 지원해야 합니다.

일반적으로 CloudFront에서 HTTP/2를 사용하는 최종 사용자와 통신하도록 구성할 경우 지연 시간이 감소합니다. HTTP/2에 대해 최적화하여 성능을 향상할 수 있습니다. 자세한 내용은 인터넷에서 "http/2 최적화"를 검색하세요.

기본 루트 객체

선택. 최종 사용자가 배포의 객체(`http://www.example.com/product-description.html`)가 아닌 배포의 루트 URL(`http://www.example.com/`)을 요청할 때 CloudFront에서 오리진(예: `index.html`)으로부터 요청할 객체입니다. 기본 루트 객체를 지정하면 배포 콘텐츠가 노출되지 않습니다.

이름의 최대 길이는 255자입니다. 이름에는 다음 문자 중 어느 것이든 포함할 수 있습니다.

- A~Z, a~z
- 0~9
- _ - . * $ / ~ " '
- &, `&`로 처리되고 반환됨

기본 루트 객체를 지정하는 경우 객체 이름만 입력합니다(예: `index.html`). 객체 이름 앞에 `/`를 추가하지 마십시오.

자세한 내용은 기본 루트 객체 지정 (p. 215) 단원을 참조하십시오.

로깅

CloudFront에서 객체에 대한 각 요청의 정보를 로깅하고 이 로그 파일을 Amazon S3 버킷에 저장할지 여부입니다. 언제든지 로깅을 활성화 또는 비활성화할 수 있습니다. 로깅을 활성화해도 추가 비용이 들지는 않지만 파일을 Amazon S3 버킷에 저장하고 액세스하는 데 일반 Amazon S3 요금이 발생합니다. 로그는 언제든지 삭제할 수 있습니다. CloudFront 액세스 로그에 대한 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

Bucket for Logs

로깅에 대해 설정을 선택한 경우, CloudFront에서 액세스 로그를 저장하려는 대상 Amazon S3 버킷입니다(예: `myawslogbucket.s3.amazonaws.com`). 로깅을 활성화한 경우, CloudFront에서는 객체에 대한 각

최종 사용자 요청의 정보를 기록하고 이 파일을 지정된 Amazon S3 버킷에 저장합니다. 언제든지 로깅을 활성화 또는 비활성화할 수 있습니다. CloudFront 액세스 로그에 대한 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

> **Note**
>
> Amazon S3 버킷 ACL을 가져오고 업데이트할 수 있는 허가가 필요하며, 해당 버킷의 S3 ACL에서 사용자에게 `FULL_CONTROL`을 부여해야 합니다. 이렇게 하면 CloudFront에서 버킷에 로그 파일을 저장할 수 있는 awsdatafeeds 계정 허가를 부여합니다. 자세한 내용은 로그 기록 및 로그 파일에 대한 액세스를 구성하는 데 필요한 권한 (p. 395) 단원을 참조하십시오.

Log Prefix

선택. 로깅에 대해 설정을 선택한 경우, CloudFront에서 이 배포에 대한 액세스 로그 파일 이름에 접두사로 지정하려는 문자열(있는 경우)을 지정합니다(예: `exampleprefix/`). 후행 슬래시(/)는 선택 사항이지만 로그 파일을 쉽게 검색할 수 있도록 넣는 것이 좋습니다. CloudFront 액세스 로그에 대한 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

Cookie Logging

CloudFront에서 쿠키를 액세스 로그에 포함하려는 경우 설정을 선택합니다. 로그에 쿠키를 포함하도록 선택한 경우, CloudFront에서는 이 배포에 대한 캐시 동작을 어떻게 구성했는지(전체 쿠키 전달, 쿠키 전달 안 함 또는 오리진에 지정한 쿠키 목록 전달)와 관계없이 모든 쿠키를 로깅합니다.

Amazon S3에서는 쿠키를 처리하지 않습니다. 따라서 배포에 Amazon EC2 또는 다른 사용자 지정 오리진 역시 포함하지 않는 한 Cookie Logging(쿠키 로깅) 값으로 Off(끄기)를 선택하는 것이 좋습니다.

쿠키에 대한 자세한 내용은 쿠키를 기반으로 콘텐츠 캐싱 (p. 198) 단원을 참조하십시오.

IPv6 사용

IPv6은 최신 버전의 IP 프로토콜입니다. 이 버전은 IPv4를 대체하며 더 큰 주소 공간을 사용합니다. CloudFront는 항상 IPv4 요청에 응답합니다. CloudFront에서 IPv4 IP 주소(예: 192.0.2.44)의 요청 및 IPv6 주소(예: 2001:0db8:85a3:0000:0000:8a2e:0370:7334)의 요청에 응답하도록 하려면 IPv6 사용을 선택합니다.

일반적으로, 콘텐츠를 액세스하려는 사용자 중에 IPv6 네트워크의 사용자가 있을 경우 IPv6를 사용하도록 설정해야 합니다. 하지만 서명된 URL이나 서명된 쿠키를 사용하여 콘텐츠에 대한 액세스를 제한하며, `IpAddress` 파라미터를 포함하는 사용자 지정 정책을 사용하여, 콘텐츠에 액세스할 수 있는 IP 주소를 제한하려는 경우에는 IPv6를 사용하도록 설정하지 마세요. 일부 콘텐츠에 대한 액세스를 IP 주소에 따라 제한하고, 다른 콘텐츠에 대한 액세스는 제한하지 않으려면(또는 액세스를 제한하되 IP 주소로 제한하지 않으려면), 두 개의 배포를 만들면 됩니다. 사용자 지정 정책을 사용하여 서명된 URL을 만드는 방법에 대한 자세한 내용은 사용자 지정 정책을 사용한 서명된 URL 생성 (p. 133)을 참조하십시오. 사용자 지정 정책을 사용하여 서명된 쿠키를 만드는 방법에 대한 자세한 내용은 사용자 지정 정책을 사용하여 서명된 쿠키 설정 (p. 148)을 참조하십시오.

Route 53 별칭 리소스 레코드 세트를 사용하여 해당 CloudFront 배포로 트래픽을 라우팅할 경우, 다음 조건이 모두 해당되면 두 번째 별칭 리소스 레코드 세트를 만들어야 합니다.

- 배포에 대해 IPv6를 사용하도록 설정함
- 객체의 URL에 대체 도메인 이름을 사용함

자세한 내용은 Amazon Route 53 개발자 안내서에서 도메인 이름을 사용하여 Amazon CloudFront 웹 배포로 트래픽 라우팅 단원을 참조하십시오.

Route 53 또는 다른 DNS 서비스를 사용하여 CNAME 리소스 레코드 세트를 만들었을 경우 아무것도 변경할 필요가 없습니다. CNAME 레코드는 최종 사용자 요청의 IP 주소 형식과 상관없이 배포로 트래픽을 라우팅합니다.

IPv6와 CloudFront 액세스 로그를 사용하도록 설정하면 `c-ip` 열에 IPv4 및 IPv6 형식의 값이 포함됩니다. 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

> **Note**
>
> IPv4가 더 우수한 사용자 경험을 제공할 것으로 추천될 경우 CloudFront는 고객에 대해 가용성을 높게 유지하기 위해 IPv4를 사용하여 최종 사용자 요청에 응답합니다. CloudFront가 IPv6를 통해 처리하는 요청의 비율을 확인하려면 배포에 대해 CloudFront 로그 기록을 활성화하여 `c-ip` 열을 구문 분석하십시오. 이 열에는 요청을 한 뷰어의 IP 주소가 들어 있습니다. 이 비율은 갈수록 커지겠지만 아직 전 세계 모든 최종 사용자 네트워크에서 IPv6가 지원되지는 않으므로 트래픽에서 상대적으로 적은 부분을 차지할 것입니다. 일부 최종 사용자 네트워크는 IPv6를 완벽하게 지원하지만, IPv6를 전혀 지원하지 않는 최종 사용자 네트워크도 있습니다. 최종 사용자 네트워크는 가정용 인터넷이나 무선 통신업체와 유사합니다.
> IPv6에 대한 지원에 대한 자세한 내용은 CloudFront FAQ를 참조하십시오. 액세스 로그 활성화에 대한 자세한 내용은 로깅 (p. 54), Bucket for Logs (p. 54) 필드와 Log Prefix (p. 55) 필드를 참조하십시오.

의견

선택. 배포를 만들 때 최대 128자의 설명을 포함할 수 있습니다. 설명은 언제든지 업데이트할 수 있습니다.

Distribution State

배포가 완료되었을 때 활성화 또는 비활성화할지 여부를 나타냅니다.

- Enabled(활성화됨)는 배포가 완료되는 대로 이 배포의 도메인 이름을 사용하는 링크를 배포할 수 있고 사용자가 콘텐츠를 가져올 수 있음을 의미합니다. 배포가 활성화되어 있으면 CloudFront에서는 해당 배포와 연결된 도메인 이름을 사용하는 콘텐츠에 대해 최종 사용자 요청을 수락하고 처리합니다.

  CloudFront 배포를 생성, 수정 또는 삭제할 경우 CloudFront 데이터베이스에 변경 사항이 전파되는 데에는 다소 시간이 소요됩니다. 배포 정보에 대한 즉각적인 요청에는 변경 사항이 반영되지 않을 수 있습니다. 전파는 대개 몇 분 내에 완료되지만, 이때 시스템에 과도한 로드가 걸리거나 네트워크 파티션이 증가할 수 있습니다.
- Disabled(비활성화됨)는 배포가 완료되어 사용할 준비가 되더라도 사용자가 이를 사용할 수 없음을 의미합니다. 배포가 비활성화되어 있으면 CloudFront에서는 해당 배포와 연결된 도메인 이름을 사용하는 최종 사용자 요청을 수락하지 않습니다. 배포의 구성을 업데이트하여 비활성 상태에서 활성 상태로 전환할 때까지는 누구도 이 배포를 사용할 수 없습니다.

원하는 만큼 배포를 비활성 상태와 활성 상태 간에 전환할 수 있습니다. 배포 구성 업데이트에 대한 프로세스를 따르세요. 자세한 내용은 배포 업데이트 (p. 59) 단원을 참조하십시오.

사용자 지정 오류 페이지 및 오류 캐싱

Amazon S3 또는 사용자 지정 오리진에서 HTTP 4xx 또는 5xx 상태 코드를 CloudFront에 반환할 때 CloudFront에서 최종 사용자에게 객체(예: HTML 파일)를 반환하도록 할 수 있습니다. 또한 CloudFront 엣지 캐시에서 오리진의 오류 응답 또는 사용자 지정 오류 페이지가 캐싱되는 시간을 지정할 수도 있습니다. 자세한 내용은 특정 HTTP 상태 코드에 대한 사용자 지정 오류 페이지 만들기 (p. 255) 단원을 참조하십시오.

> **Note**
>
> 다음 값은 Create Distribution 마법사에 포함되어 있지 않으므로 배포를 업데이트할 때만 사용자 지정 오류 페이지를 구성할 수 있습니다.

Error Code

CloudFront에서 사용자 지정 오류 페이지를 반환하려는 HTTP 상태 코드입니다. CloudFront에서 캐싱하는 HTTP 상태 코드의 일부 또는 전부에 대한(또는 상태 코드가 없는) 사용자 지정 오류 페이지를 반환하도록 CloudFront를 구성할 수 있습니다.

Response Page Path

403 등의 Error Code(오류 코드)로 지정한 HTTP 상태 코드가 오리진에서 반환될 때 CloudFront에서 최종 사용자에게 반환하는 사용자 지정 오류 페이지의 경로(예: `/4xx-errors/403-forbidden.html`)입니다. 객체와 사용자 지정 오류 페이지를 서로 다른 위치에 저장하려는 경우, 배포에는 다음을 충족하는 캐시 동작이 포함되어야 합니다.

- 경로 패턴의 값이 사용자 지정 오류 메시지와 일치합니다. 예를 들어, `/4xx-errors`라는 이름의 디렉터리에 있는 Amazon S3 버킷에 4xx 오류에 대한 사용자 지정 오류 페이지를 저장했다고 가정합니다. 배포에는 사용자 지정 오류 페이지에 대한 요청을 해당 위치로 라우팅하는 경로 패턴(예: /4xx-errors/*)의 캐시 동작이 포함되어야 합니다.
- Origin(오리진)의 값은 사용자 지정 오류 페이지를 포함한 오리진에 대한 Origin ID(오리진 ID)의 값을 지정합니다.

Response Code

CloudFront에서 사용자 지정 오류 페이지와 함께 최종 사용자에게 반환하려는 HTTP 상태 코드입니다.

Error Caching Minimum TTL

CloudFront에서 오리진 서버로부터 오류 응답을 캐싱하려는 최소 시간입니다.

제한 사항

선택한 국가의 사용자가 콘텐츠에 액세스하는 것을 방지해야 할 경우, 지정된 화이트리스트 국가의 사용자에게 콘텐츠 액세스를 허용하거나 지정된 블랙리스트 국가의 사용자에게 콘텐츠 액세스를 허용하지 않도록 CloudFront 배포를 구성할 수 있습니다. 자세한 내용은 콘텐츠의 지리적 배포 제한 (p. 183) 단원을 참조하십시오.

> **Note**
>
> 다음 값은 Create Distribution 마법사에 포함되어 있지 않으므로 배포를 업데이트할 때만 지리적 제한을 구성할 수 있습니다.

Enable Geo Restriction

특정 국가의 사용자가 콘텐츠에 액세스하는 것을 차단할지 여부입니다. 지리적 제한을 구성하는 데 드는 추가 요금은 없습니다.

Restriction Type

콘텐츠에 액세스할 수 있는 사용자가 위치한 국가를 지정하려는 방법입니다.

- Whitelist(화이트리스트): Countries(국가) 목록에 콘텐츠에 대한 액세스를 허용하려는 모든 국가를 포함합니다.
- Blacklist(블랙리스트): Countries(국가) 목록에 콘텐츠에 대한 액세스를 차단하려는 모든 국가를 포함합니다.

Countries

화이트리스트 또는 블랙리스트에 추가하려는 국가입니다. 국가를 추가하려면 왼쪽 목록에서 해당 국가를 선택하고 추가를 선택합니다. 다음을 참조하십시오.

- 연속된 여러 국가를 추가하려면 첫 번째 국가를 선택한 후 Shift 키를 누른 상태로 마지막 국가를 선택하고 추가를 선택합니다.

- 비연속적인 여러 국가를 추가하려면 첫 번째 국가를 선택한 후 Ctrl 키를 누른 상태로 나머지 국가들을 선택하고 추가를 선택합니다.
- 왼쪽 목록에서 국가를 찾으려면 국가의 전체 이름에서 처음 몇 글자를 입력합니다.
- 각 국가 이름 앞의 두 글자 코드가 CloudFront API를 사용하여 배포를 생성 또는 업데이트하려는 경우 입력할 값입니다. ISO(국제 표준화 기구) 국가 코드가 사용됩니다. 사용이 간편한 목록, 코드와 국가 이름으로 분류 가능한 목록에 대해서는 Wikipedia 항목 ISO 3166-1 alpha-2를 참조하십시오.

CloudFront가 콘솔에 표시하는 값

새 배포를 생성하거나 기존 배포를 업데이트할 때, CloudFront에서는 다음 정보를 CloudFront 콘솔에 표시합니다.

> **Note**
>
> 활성화된 신뢰할 수 있는 서명자, 활성 CloudFront 키 페어를 보유하고 있으며 유효한 서명된 URL을 만드는 데 사용할 수 있는 AWS 계정은 현재 CloudFront 콘솔에 표시되지 않습니다.

Distribution ID(General 탭)

CloudFront API를 사용하여 배포에 대한 작업을 수행할 경우, 배포 ID를 사용하여 대상 배포를 지정합니다(예: `EDFDVBD6EXAMPLE`). 배포의 배포 ID는 변경할 수 없습니다.

Distribution Status(General 탭)

배포에 대해 가능한 상태 값은 다음 표에 나와 있습니다.

값	설명
InProgress	배포가 생성 중이거나 업데이트 중이며, 변경 사항이 엣지 서버에 아직 완전히 전파되지 않았습니다.
배포 완료	배포의 생성 또는 업데이트가 완료되어 변경 사항이 CloudFront 시스템 전반에 완전히 전파되었습니다.

> **Note**
>
> 배포의 상태가 배포 완료가 되어야 하는 것에 추가로, 사용자가 CloudFront를 사용하여 콘텐츠에 액세스할 수 있으려면 먼저 배포를 활성화해야 합니다. 자세한 내용은 Distribution State (p. 56) 단원을 참조하십시오.

Last Modified(General 탭)

배포가 마지막으로 수정된 날짜 및 시간을 ISO 8601 형식을 사용하여 나타냅니다(예: 2012-05-19T19:37:58Z). 자세한 내용은 http://www.w3.org/TR/NOTE-datetime을 참조하십시오.

Domain Name(General 탭)

객체에 대한 링크에 배포의 도메인 이름을 사용합니다. 예를 들어, 배포의 도메인 이름이 `d111111abcdef8.cloudfront.net`인 경우, `/images/image.jpg`에 대한 링크는 `http://d111111abcdef8.cloudfront.net/images/image.jpg`가 됩니다. 배포에 대한 CloudFront 도메인 이름은 변경할 수 없습니다. 객체에 대한 링크의 CloudFront URL에 대한 자세한 내용은 CloudFront에서 파일에 대한 URL 형식 사용자 지정 (p. 79) 단원을 참조하십시오.

하나 이상의 대체 도메인 이름(CNAME)을 지정한 경우, 객체에 대한 링크에 CloudFront 도메인 이름을 사용하는 대신 자체 도메인 이름을 사용할 수 있습니다. CNAME에 대한 자세한 내용은 Alternate Domain Names (CNAMEs) (p. 51)을 참조하십시오.

> **Note**
>
> CloudFront 도메인 이름은 고유합니다. 즉, 배포의 도메인 이름은 이전 배포에 사용된 적이 없고 이후 다른 배포에 재사용되지도 않습니다.

배포 테스트

배포를 생성하면 CloudFront에서는 오리진 서버의 위치를 알 수 있으며, 사용자는 배포와 연결된 도메인 이름을 알 수 있습니다. CloudFront 도메인 이름을 사용하여 객체에 대한 링크를 만들 수 있으며, CloudFront에서는 웹 페이지 또는 애플리케이션에 객체를 제공합니다.

> **Note**
>
> 링크를 테스트하기 전에 배포의 상태가 배포 완료로 변경될 때까지 기다려야 합니다.

웹 배포에 객체에 대한 링크를 생성하려면

1. 다음 HTML 코드를 새 파일에 복사하고 *domain-name*과 *object-name*을 각각 배포의 도메인 이름과 객체의 이름으로 대체합니다.

   ```
   <html>
   <head>My CloudFront Test</head>
   <body>
   <p>My text content goes here.</p>
   <p><img src="http://domain-name/object-name" alt="my test image"
   </body>
   </html>
   ```

   예를 들어, 도메인 이름이 `d111111abcdef8.cloudfront.net`이고 객체가 `image.jpg`인 경우 링크의 URL은 다음과 같습니다.

   `http://d111111abcdef8.cloudfront.net/image.jpg`를 선택하십시오.

   객체가 오리진 서버의 폴더에 있는 경우, URL에 이 폴더도 포함해야 합니다. 예를 들어, image.jpg가 오리진 서버의 이미지 폴더에 있으면 URL은 다음과 같습니다.

   `http://d111111abcdef8.cloudfront.net/images/image.jpg`
2. 파일 이름 확장명이 .html인 파일에 HTML 코드를 저장합니다.
3. 브라우저에서 웹 페이지를 열어 객체를 볼 수 있는지 확인합니다.

브라우저가 엣지에서 제공한 이미지 파일이 포함된 페이지를 반환합니다. 이 엣지는 CloudFront에서 객체를 서비스하기에 적합한 것으로 확인된 위치입니다.

배포 업데이트

CloudFront 콘솔에서 AWS 계정과 연결된 CloudFront 배포를 확인하고, 배포에 대한 설정을 보며, 대부분의 설정을 업데이트할 수 있습니다. 변경한 설정은 배포가 AWS 엣지 로케이션에 전파되어야 적용됩니다.

CloudFront 배포를 업데이트하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 배포의 ID를 선택합니다. 이 목록에는 CloudFront 콘솔에 로그인하는 데 사용한 AWS 계정과 연결된 배포가 모두 포함되어 있습니다.

3. 배포에 대한 설정을 보거나 수정하려면 Distribution Settings(배포 설정) 탭을 선택합니다.
4. 업데이트하려면 다음 중 하나를 수행합니다.

   a. 일반 설정을 업데이트하려면 편집을 선택합니다. 또는 업데이트하려는 설정의 오리진 탭 또는 동작 탭을 선택합니다.
   b. RTMP 배포 설정의 경우 편집을 선택한 후 값을 업데이트합니다.

      필드에 대한 자세한 내용은 RTMP 배포의 생성 또는 업데이트 시 지정하는 값 (p. 272) 단원을 참조하십시오.
5. 업데이트한 후 변경한 내용을 저장하려면 예, 편집합니다.를 선택합니다. 필드에 대한 자세한 내용은 다음 주제를 참조하십시오.

   - 일반 설정: 배포 세부 정보 (p. 51)
   - 오리진 설정: 오리진 설정 (p. 38)
   - 캐시 동작 설정: 캐시 동작 설정 (p. 43)
6. 배포에서 오리진을 삭제하려면 다음과 같이 합니다.

   a. 동작을 선택하고 해당 오리진과 연결된 기본 캐시 동작을 다른 오리진으로 옮겼는지 확인합니다.
   b. 오리진을 선택한 후 오리진을 선택합니다.
   c. 삭제를 선택합니다.

CloudFront API를 사용하여 배포를 업데이트할 수도 있습니다.

- 배포를 업데이트하려면 Amazon CloudFront API Reference에서 UpdateDistribution을 참조하십시오.

  Important

  배포를 업데이트하는 경우에는 배포를 만들 때는 필요 없었던 여러 개의 추가 필드가 필요합니다. 배포를 생성 또는 업데이트하는 경우 필요한 필드에 대한 요약 설명은 배포 생성 및 배포 업데이트에 필요한 필드 (p. 32) 단원을 참조하십시오. CloudFront API를 사용하여 배포를 업데이트하는 경우 필수 필드를 모두 포함하려면 Amazon CloudFront API Reference의 UpdateDistribution에 설명된 단계를 따르십시오.

배포 구성에 대한 변경 내용을 저장하면 CloudFront에서 이러한 변경 내용을 모든 엣지 로케이션으로 전파하기 시작합니다. 구성이 엣지 로케이션에서 업데이트될 때까지 CloudFront는 이전 구성에 따라 해당 위치에서 콘텐츠를 계속 제공합니다. 구성이 엣지 로케이션에서 업데이트된 후에는 CloudFront가 즉시 새 구성에 따라 해당 위치에서 콘텐츠를 제공하기 시작합니다.

변경 내용이 모든 엣지 로케이션으로 즉시 전파되지는 않습니다. 전파가 완료되면 배포의 상태가 진행 중에서 배포 완료로 변경됩니다. CloudFront가 변경 내용을 전파하고 있는 동안은 지정된 엣지 로케이션이 이전 구성과 새 구성 중 어느 것에 기초하여 콘텐츠를 제공하는지 확인할 수 없습니다.

배포 삭제

더 이상 배포를 사용하고 싶지 않으면 CloudFront 콘솔이나 CloudFront API를 사용하여 삭제할 수 있습니다.

배포를 삭제하려면 먼저 비활성화해야 하며, 이를 위해서는 배포 업데이트 권한이 필요하다는 점에 유의하십시오. CloudFront로 작업할 권한 설정(UpdateDistribution 및 DeleteDistribution 권한 설정 포함)에 대한 자세한 내용은 고객 관리형 정책 예 (p. 437) 단원을 참조하십시오.

Note

대체 도메인 이름이 연결된 배포를 비활성화하면, 다른 배포에 동일한 도메인과 일치하는 와일드카드(*)가 있는 대체 도메인 이름(예: *.example.com)이 있더라도 CloudFront가 해당 도메인 이름(예: www.example.com)에 대한 트래픽 수신을 중지합니다.

CloudFront 배포를 삭제하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. CloudFront 콘솔의 오른쪽 창에서 삭제할 배포를 찾습니다.
3. 상태 열의 값이 비활성화됨이면 7단계로 이동합니다.

   상태 값이 활성화됨이고 상태의 값이 배포 완료이면 삭제하기 전에 4단계를 계속 진행하여 배포를 비활성화합니다.

   상태의 값이 활성이고 상태의 값이 진행 중이면 상태가 배포 완료로 변경될 때까지 기다립니다. 그런 다음 삭제하기 전에 4단계를 계속 진행하여 배포를 비활성화합니다.
4. CloudFront 콘솔의 오른쪽 창에서 삭제할 배포에 해당하는 확인란을 선택합니다.
5. 비활성화를 선택하여 배포를 비활성화하고 예, 비활성화를 선택하여 확인합니다. 그런 다음 닫기를 선택합니다.

   Note

   CloudFront는 이 변경 사항을 모든 엣지 로케이션에 전파해야 하므로, 업데이트가 완료되고 배포를 삭제할 수 있으려면 최대 90분이 걸릴 수 있습니다.
6. 상태 열의 값이 비활성으로 즉시 변경됩니다. 상태 열의 값이 배포 완료로 변경될 때까지 기다립니다.
7. 삭제할 배포에 해당하는 확인란을 선택합니다.
8. 삭제를 선택하고 예, 삭제를 선택하여 확인합니다. 그런 다음 닫기를 클릭합니다.

   Note

   방금 배포를 비활성화됨으로 표시한 경우, CloudFront가 해당 변경 사항을 엣지 로케이션에 전파하려면 몇 분이 더 필요할 수 있습니다. 전파가 완료될 때까지 삭제 옵션은 사용할 수 없습니다.

CloudFront API를 사용하여 배포를 삭제할 수도 있습니다.

- 배포를 삭제하려면 Amazon CloudFront API Reference의 DeleteDistribution을 참조하십시오.
- RTMP 배포를 삭제하려면 Amazon CloudFront API Reference의 DeleteStreamingDistribution을 참조하십시오.

웹 배포를 위해 Amazon S3 오리진, MediaPackage 채널, 사용자 지정 오리진 사용

배포를 생성할 때 CloudFront가 엣지 로케이션에 배포할 파일에 대한 요청을 보내는 위치를 지정합니다. CloudFront는 오리진으로 여러 AWS 리소스를 사용하는 것을 지원합니다. 예를 들어 Amazon S3 버킷이나 MediaStore 컨테이너, MediaPackage 채널 또는 사용자 지정 오리진(Amazon EC2 인스턴스 또는 HTTP 웹 서버 등)을 지정할 수 있습니다.

주제

- Amazon S3 버킷을 오리진으로 사용 (p. 62)
- 오리진에 대한 웹 사이트 엔드포인트로 구성된 Amazon S3 버킷 사용 (p. 62)
- 오리진에 대한 MediaStore 컨테이너 또는 MediaPackage 채널 사용 (p. 63)
- Amazon EC2 또는 기타 사용자 지정 오리진 사용 (p. 63)
- CloudFront 오리진 그룹 사용 (p. 64)
- Amazon S3에서 이미 콘텐츠를 배포하고 있는 경우 CloudFront 추가 (p. 64)

- Amazon S3 버킷을 다른 리전으로 이동 (p. 66)

Amazon S3 버킷을 오리진으로 사용

배포에 대한 오리진으로 Amazon S3를 사용하는 경우, CloudFront에서 제공하려는 모든 객체를 Amazon S3 버킷에 배치합니다. Amazon S3 콘솔이나 API, 또는 타사 도구 등 Amazon S3에서 지원되는 모든 방식을 사용하여 객체를 Amazon S3에 넣을 수 있습니다. 다른 Amazon S3 버킷과 마찬가지로 버킷에 계층을 만들어 객체를 저장할 수 있습니다.

기존 Amazon S3 버킷을 CloudFront 오리진 서버로 사용하더라도 버킷은 어떠한 방식으로도 변경되지 않습니다. 평상시처럼 Amazon S3 객체를 저장 및 액세스할 용도로 표준 Amazon S3 가격에 버킷을 계속해서 사용할 수 있습니다. 버킷에 객체를 저장하기 위해서는 정규 Amazon S3 요금이 발생합니다. CloudFront 사용을 위한 요금에 대한 자세한 내용은 콘솔의 CloudFront 보고서 (p. 369) 단원을 참조하십시오.

> **Important**
>
> 버킷을 CloudFront와 함께 사용하려면 DNS 명명 요구 사항을 준수하는 이름을 사용해야 합니다. 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 버킷 규제 및 제한 단원을 참조하십시오.

CloudFront가 객체를 가져오고자 하는 Amazon S3 버킷을 지정할 때 다음 형식을 사용하여 버킷에 액세스하는 것이 좋습니다.

```
bucket-name.s3.region.amazonaws.com
```

다른 옵션에서는 다음의 더 일반적인 형식을 사용할 수 있지만, 2019년 이후 출시된 리전에서는 이 형식을 사용할 수 없습니다.

```
bucket-name.s3.amazonaws.com
```

이 형식으로 버킷 이름을 지정할 경우 다음 CloudFront 기능을 사용할 수 있습니다.

- SSL을 사용하여 Amazon S3 버킷과 통신하도록 CloudFront를 구성합니다. 자세한 내용은 CloudFront에서 HTTPS 사용 (p. 92) 단원을 참조하십시오.
- 원본 액세스 ID를 사용하여 사용자에게 Amazon S3 URL이 아닌 CloudFront URL을 사용하여 콘텐츠에 액세스하도록 요구합니다. 자세한 내용은 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오.
- `POST` 및 `PUT` 요청을 CloudFront에 제출하여 버킷의 콘텐츠를 업데이트합니다. 자세한 내용은 HTTP 메소드 (p. 232) 주제에서 CloudFront에서 요청을 처리하고 Amazon S3 오리진 서버에 요청을 전달하는 방법 (p. 230) 단원을 참조하십시오.

다음 형식을 사용하여 버킷을 지정하지 마세요.

- Amazon S3 경로 스타일: `s3.amazonaws.com/bucket-name`
- Amazon S3 CNAME(있는 경우)

오리진에 대한 웹 사이트 엔드포인트로 구성된 Amazon S3 버킷 사용

CloudFront를 사용하여 사용자 지정 오리진으로서 웹 사이트 엔드포인트로 구성되는 Amazon S3 버킷을 설정할 수 있습니다.

- CloudFront 배포를 구성할 때 오리진의 경우 버킷에 대한 Amazon S3 정적 웹 사이트 호스팅 엔드포인트를 입력합니다. 이 값은 Amazon S3 콘솔에서 속성 페이지의 정적 웹 사이트 호스팅 아래에 나타납니다. 예:

```
https://bucket-name.s3-website.region.amazonaws.com
```

Amazon S3 정적 웹 사이트 엔드포인트 지정에 대한 자세한 내용은 Amazon S3 설명서의 웹 사이트 엔드포인트 단원을 참조하십시오.

오리진으로서 이 형식으로 된 버킷 이름을 지정할 경우 Amazon S3 리디렉션과 Amazon S3 사용자 지정 오류 문서를 사용할 수 있습니다. Amazon S3 기능에 대한 자세한 내용은 Amazon S3 설명서를 참조하십시오. (CloudFront에서는 사용자 지정 오류 페이지도 제공합니다. 자세한 내용은 특정 HTTP 상태 코드에 대한 사용자 지정 오류 페이지 만들기 (p. 255) 단원을 참조하십시오.

CloudFront 오리진 서버로서 Amazon S3 버킷을 사용해도 변경되지 않습니다. 평소처럼 계속해서 사용할 수 있으며 정규 Amazon S3 요금이 발생합니다. CloudFront 사용을 위한 요금에 대한 자세한 내용은 콘솔의 CloudFront 보고서 (p. 369) 단원을 참조하십시오.

> **Note**
>
> CloudFront API를 사용하여 웹 사이트 엔드포인트로 구성되는 Amazon S3 버킷 포함 배포를 생성하는 경우 웹 사이트가 Amazon S3 버킷에서 호스팅된 경우에도 `CustomOriginConfig`를 사용하여 구성해야 합니다. CloudFront API를 사용한 배포 만들기에 대한 자세한 내용은 Amazon CloudFront API Reference의 CreateDistribution 단원을 참조하십시오.

오리진에 대한 MediaStore 컨테이너 또는 MediaPackage 채널 사용

CloudFront를 사용하여 비디오를 스트리밍하려면 MediaStore 컨테이너로 구성된 Amazon S3 버킷을 설정하거나, MediaPackage를 사용하여 채널 및 엔드포인트를 생성할 수 있습니다. 그런 다음 CloudFront에서 배포를 만들고 구성하여 비디오를 스트리밍합니다.

자세한 내용 및 단계별 지침은 다음 단원을 참조하십시오.

- AWS Elemental MediaStore를 오리진으로 사용하여 비디오 제공 (p. 264)
- AWS Elemental MediaPackage를 사용하여 포맷된 라이브 비디오 제공 (p. 265)

Amazon EC2 또는 기타 사용자 지정 오리진 사용

사용자 지정 오리진은 웹 서버와 같은 HTTP 서버입니다. HTTP 서버는 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스나 비공개로 관리하는 HTTP 서버가 될 수 있습니다. 웹 사이트 엔드포인트로서 구성된 Amazon S3 오리진은 또한 사용자 지정 오리진으로 간주됩니다.

고유 HTTP 서버인 사용자 지정 오리진을 사용하는 경우 서버의 DNS 이름, HTTP 및 HTTPS 포트, 오리진에서 객체를 가져올 때 CloudFront에서 사용하려는 프로토콜을 지정합니다.

사용자 지정 오리진을 사용할 경우 대부분의 CloudFront 기능이 지원되며 단, 다음 예외 사항이 적용됩니다.

- RTMP 배포 - 지원되지 않음
- 비공개 콘텐츠 - 서명된 URL을 사용하여 사용자 지정 오리진으로부터 콘텐츠를 배포할 수는 있지만, CloudFront에서 사용자 지정 오리진을 액세스하려면 오리진이 공개적으로 액세스 가능한 상태여야 합니다. 자세한 내용은 서명된 URL과 서명된 쿠키를 사용하여 프라이빗 콘텐츠 제공 (p. 115) 단원을 참조하십시오.

CloudFront를 통한 Amazon EC2 인스턴스 및 기타 사용자 지정 오리진 사용에 대한 이러한 지침을 따르십시오.

- 동일한 CloudFront 오리진에 대한 콘텐츠를 서비스하는 모든 서버의 동일한 콘텐츠를 호스팅 및 서비스합니다. 자세한 내용은 오리진 설정 (p. 38) 주제에서 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.
- `X-Amz-Cf-Id` 헤더 항목을 모든 서버에 로깅합니다. CloudFront에서는 디버깅을 위해 이 정보가 필요합니다.
- 사용자 지정 오리진이 수신 대기하는 HTTP 및 HTTPS 포트에 대한 액세스 요청을 제한합니다.
- 구현 시 모든 서버의 클록을 동기화합니다. CloudFront는 액세스 로그 및 보고용으로 서명된 URL 및 서명된 쿠키에 대해 UTC(협정 세계시)를 사용합니다. 추가로 CloudWatch 지표를 사용하여 CloudFront 활동을 모니터링하는 경우 CloudWatch는 또한 UTC를 사용합니다.
- 이중화 서버를 사용하여 오류를 처리합니다.
- 프라이빗 콘텐츠를 제공하기 위한 사용자 지정 오리진 사용에 대한 자세한 내용은 사용자 지정 오리진의 파일에 대한 액세스 제한 (p. 117)을 참조하십시오.
- 요청 및 응답 동작과 지원되는 HTTP 상태 코드에 대한 자세한 내용은 요청 및 응답 동작 (p. 230) 단원을 참조하십시오.

Amazon EC2를 사용자 지정 오리진으로 사용하는 경우 다음을 수행하는 것이 좋습니다.

1. 웹 서버용 소프트웨어를 자동으로 설치하는 Amazon 머신 이미지를 사용합니다. 자세한 내용은 Amazon EC2 설명서를 참조하십시오.
2. Elastic Load Balancing 로드 밸런서를 사용하여 여러 Amazon EC2 인스턴스 간 트래픽을 처리하고 Amazon EC2 인스턴스에 대한 변경으로부터 애플리케이션을 격리합니다. 예를 들어, 로드 밸런서를 사용하는 경우 애플리케이션을 변경하지 않고도 Amazon EC2 인스턴스를 추가 및 삭제할 수 있습니다. 자세한 내용은 Elastic Load Balancing 설명서를 참조하십시오.
3. CloudFront 배포를 생성할 때 오리진 서버의 도메인 이름으로 로드 밸런서의 URL을 지정합니다. 자세한 내용은 배포 만들기 (p. 35) 단원을 참조하십시오.

CloudFront 오리진 그룹 사용

예를 들어 고가용성이 필요할 때 시나리오에 대한 오리진 장애 조치를 구성하려면 CloudFront 오리진에 대한 오리진 그룹을 지정할 수 있습니다. 오리진 장애 조치를 사용하여 CloudFront에 대한 기본 오리진과, 기본 오리진이 특정 HTTP 상태 코드 실패 응답을 반환할 때 CloudFront가 자동으로 전환하는 두 번째 오리진을 지정합니다.

오리진 그룹을 설정하는 단계 및 자세한 내용은 CloudFront 오리진 장애 조치를 통한 고가용성 최적화 (p. 209) 단원을 참조하십시오.

Amazon S3에서 이미 콘텐츠를 배포하고 있는 경우 CloudFront 추가

Amazon S3 버킷에 객체를 저장하는 경우 사용자에게 S3에서 객체를 직접 가져오게 할 수 있고, 아니면 S3에서 객체를 가져와 사용자들에게 배포하도록 CloudFront를 구성할 수 있습니다.

> **Note**
>
> Amazon S3 버킷을 웹 사이트 엔드포인트로 구성하는 경우를 포함하여 CloudFront를 통해 오리진에 대한 Amazon S3 버킷 사용에 대한 자세한 내용은 웹 배포를 위해 Amazon S3 오리진, MediaPackage 채널, 사용자 지정 오리진 사용 (p. 61) 단원을 참조하십시오.

사용자가 객체에 자주 액세스하는 경우 CloudFront를 사용하는 것이 더 비용 효율적일 수 있습니다. 왜냐하면 사용량이 더 많은 경우에는 CloudFront 데이터 전송 가격이 Amazon S3 데이터 전송 가격보다 저렴하기 때문입니다. 또한 CloudFront를 함께 사용할 경우 객체가 사용자 측에 더 가깝게 저장되므로 다운로드 속도도 Amazon S3만 사용할 때보다 더 빠릅니다.

> **Note**
>
> CloudFront에서 Amazon S3 cross-origin 리소스 공유 설정을 지키도록 하려는 경우 `Origin` 헤더를 Amazon S3으로 전달하도록 CloudFront를 구성합니다. 자세한 내용은 요청 헤더에 따라 콘텐츠 캐싱 (p. 200) 단원을 참조하십시오.

현재 Amazon S3 버킷(예: MyAWSBucket.s3.us-west-2.amazonaws.com)을 사용하지 않고 고유 도메인 이름(예: example.com)을 사용하여 Amazon S3 버킷에서 콘텐츠를 직접 배포하는 경우, 다음 절차를 이용해 중단 없이 CloudFront를 추가할 수 있습니다.

Amazon S3에서 콘텐츠를 이미 배포하고 있는 경우 CloudFront를 추가하려면

1. 다음 절차 중 하나를 수행하여 CloudFront 배포를 생성합니다.

   - 배포 만들기 단계(개요) (p. 34)
   - RTMP를 사용한 미디어 파일 스트리밍용 작업 목록 (p. 271)

   배포를 생성할 때 Amazon S3 버킷의 이름을 오리진 서버로 지정합니다.

   > **Important**
   >
   > 버킷을 CloudFront와 함께 사용하려면 DNS 명명 요구 사항을 준수하는 이름을 사용해야 합니다. 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 Bucket Restrictions and Limitations를 참조하십시오.

   Amazon S3에서 CNAME을 사용하려는 경우 배포에 대한 CNAME도 지정합니다.
2. Amazon S3 버킷 안에 공개 읽기 가능 객체의 링크를 포함하는 테스트 웹 페이지를 생성하고 해당 링크를 테스트합니다. 이 초기 테스트에서는 객체 URL에서 배포의 CloudFront 도메인 이름을 사용합니다 (예: `http://d111111abcdef8.cloudfront.net/images/image.jpg`).

   CloudFront URL의 형식에 대한 자세한 내용은 CloudFront에서 파일에 대한 URL 형식 사용자 지정 (p. 79) 단원을 참조하십시오.
3. Amazon S3 CNAME을 사용할 경우 애플리케이션은 Amazon S3 버킷에서 객체를 참조할 때 버킷 이름 (예: *`aws-s3-bucket1`*.s3.amazonaws.com) 대신에 도메인 이름(예: example.com)을 사용합니다. 객체를 참조할 때 배포의 CloudFront 도메인 이름(예: d111111abcdef8.cloudfront.net) 대신에 고유 도메인 이름을 계속 사용하려면 설정을 해당 DNS 서비스 공급자로 업데이트해야 합니다.

   Amazon S3 CNAME이 작동하려면 DNS 서비스 공급자에게 고유한 도메인에 대한 CNAME 리소스 레코드 세트가 반드시 있어야 합니다. 이 리소스 레코드 세트는 현재 도메인에 대한 쿼리를 Amazon S3 버킷으로 라우팅합니다. 예를 들어 사용자가 이 객체를 요청하는 경우:

   ```
   http://example.com/images/image.jpg
   ```

   요청이 자동으로 재라우팅되어 사용자에게 표시되는 객체는 다음과 같습니다.

   ```
   http://aws-s3-bucket1.s3.amazonaws.com/images/image.jpg
   ```

   쿼리를 Amazon S3 버킷 대신에 CloudFront 배포로 라우팅하려면 DNS 서비스 공급자가 제공하는 방법을 사용하여 도메인의 CNAME 리소스 레코드 세트를 업데이트해야 합니다. 이 업데이트된 CNAME 레코드가 DNS 쿼리를 고유 도메인에서 배포의 CloudFront 도메인 이름으로 리디렉션하기 시작합니다. 자세한 내용은 DNS 서비스 공급자가 제공하는 설명서를 참조하십시오.

   > **Note**
   >
   > Route 53을 DNS 서비스로 사용하는 경우 CNAME 리소스 레코드 세트 또는 별칭 리소스 레코드 세트를 사용할 수 있습니다. 리소스 레코드 세트 편집에 대한 자세한 내용은 리소스 레코드 세트 편집 단원을 참조하십시오. 별칭 리소스 레코드 세트에 대한 자세한 내용은 별칭 및 비-별칭 리소스 레코드 세트 간의 선택 단원을 참조하십시오. 두 주제는 모두 Amazon Route 53 개발자 안내서에 있습니다.

CloudFront에 CNAME 사용에 대한 자세한 내용은 대체 도메인 이름(CNAME)을 추가하여 파일에 대해 사용자 지정 URL 사용 (p. 66) 단원을 참조하십시오.

일반적으로 더 빠를 수도 있지만 CNAME 리소스 레코드 세트를 업데이트한 후 DNS 시스템 전체에 변경 사항이 전파되는 데는 최대 72시간이 걸릴 수 있습니다. 이 시간 동안은 일부 콘텐츠 요청이 Amazon S3 버킷으로 계속 라우팅되며 그 밖의 요청은 CloudFront로 라우팅됩니다.

Amazon S3 버킷을 다른 리전으로 이동

Amazon S3을 CloudFront 배포에 대한 오리진으로 사용하고 있고 버킷을 다른 리전으로 옮기는 경우, 다음 두 조건이 참일 때 CloudFront가 그 레코드를 업데이트하여 리전 변경을 포함하는 데 최대 1시간이 걸릴 수 있습니다.

- CloudFront 원본 액세스 ID(OAI)를 사용하여 버킷에 대한 액세스를 제한하고 있습니다.
- 인증을 위해 서명 버전 4가 필요한 Amazon S3 리전으로 버킷을 옮깁니다.

OAI를 사용할 때 CloudFront는 그 리전(다른 값들 중에서)을 사용해 버킷에서 객체를 요청할 때 사용하는 서명을 계산합니다. OAI에 대한 자세한 내용은 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오. Amazon S3 리전 목록과 그 리전에서 지원되는 서명 버전은 Amazon Web Services 일반 참조의 "리전 및 엔드포인트" 장에서 Amazon Simple Storage Service(Amazon S3)를 참조하십시오.

CloudFront의 레코드에 대한 더 빠른 업데이트를 강제하려면, CloudFront 콘솔에서 일반 탭의 설명을 업데이트하는 등의 작업을 통해 CloudFront 배포를 업데이트할 수 있습니다. 배포를 업데이트하는 경우, CloudFront는 버킷이 있는 리전을 즉시 확인합니다. 한편, 변경 사항을 모든 엣지 로케이션으로 전파하는 데 걸리는 시간은 15분 미만입니다.

대체 도메인 이름(CNAME)을 추가하여 파일에 대해 사용자 지정 URL 사용

CloudFront에서 CNAME이라고 하는 대체 도메인 이름을 사용하면 파일 링크용으로 CloudFront에서 배포에 배정하는 도메인 이름 대신에 고유의 도메인 이름(예: `www.example.com`)이 사용됩니다. 웹 및 RTMP 배포 모두에서 대체 도메인 이름이 지원됩니다.

배포를 생성하는 경우 CloudFront는 배포용 도메인 이름을 반환합니다. 예를 들면 다음과 같습니다.

```
d111111abcdef8.cloudfront.net
```

파일에 CloudFront 도메인 이름을 사용하는 경우 `/images/image.jpg`라는 파일 URL은 다음과 같습니다.

```
http://d111111abcdef8.cloudfront.net/images/image.jpg
```

`cloudfront.net` 도메인 이름 대신에 `www.example.com` 같은 고유 도메인 이름을 사용하려는 경우, 대체 도메인 이름(예: `www.example.com`)을 배포에 추가할 수 있습니다. 그런 다음 다음 URL을 사용하여 `/images/image.jpg`를 볼 수 있습니다.

```
http://www.example.com/images/image.jpg
```

주제

- 대체 도메인 이름 추가 (p. 67)
- 대체 도메인 이름을 다른 CloudFront 배포로 이동 (p. 69)
- 대체 도메인 이름 제거 (p. 72)
- CloudFront에 추가하는 대체 도메인 이름에서 와일드카드 사용 (p. 73)

- 대체 도메인 이름 사용과 관련된 요구 사항 (p. 73)
- 대체 도메인 이름에 대한 제한 (p. 74)

대체 도메인 이름 추가

다음 작업 목록은 CloudFront 콘솔을 사용하여 배포에 대체 도메인 이름을 추가하여 링크에서 CloudFront 도메인 이름 대신에 고유의 도메인 이름을 사용할 수 있도록 하는 방법을 설명합니다. CloudFront API를 사용한 배포 업데이트에 대한 자세한 내용은 배포 작업 (p. 31) 단원을 참조하십시오.

> **Note**
>
> 최종 사용자가 대체 도메인 이름에 HTTPS를 사용하도록 하려면 대체 도메인 이름과 HTTPS 사용 (p. 102) 단원을 참조하십시오.

시작하기 전에: 대체 도메인 이름을 추가하도록 배포를 업데이트하기 전에 다음을 수행해야 합니다.

- 도메인 이름을 Route 53 또는 다른 도메인 공급자에 등록합니다.
- 배포에 사용할 도메인 이름이 포함된 인증된 인증 기관(CA)의 인증서를 CloudFront에 추가하여 해당 도메인을 사용할 권한을 검증합니다. 자세한 내용은 대체 도메인 이름 사용과 관련된 요구 사항 (p. 73) 단원을 참조하십시오.

대체 도메인 이름 추가

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 업데이트하려는 배포의 ID를 선택합니다.
3. [General] 탭에서 [Edit]를 선택합니다.
4. 다음 값을 업데이트합니다.

   Alternate Domain Names (CNAMEs)

   대체 도메인 이름을 추가합니다. 도메인 이름을 쉼표로 구분하거나 각각의 이름을 새 줄에 입력합니다.

   SSL 인증서(웹 배포만 해당)

   다음 설정을 선택합니다.

   - Use HTTPS(HTTPS 사용) – 사용자 지정 SSL 인증서를 선택하고 목록에서 인증서를 선택합니다. 이 목록에는 AWS Certificate Manager(ACM)에서 프로비저닝된 인증서, 다른 CA에서 구입하여 ACM에 업로드한 인증서, 다른 CA에서 구입하여 IAM 인증서 스토어에 업로드한 인증서 등이 포함될 수 있습니다.

     IAM 인증서 스토어에 인증서를 업로드했는데 목록에 나타나지 않으면 SSL/TLS 인증서 가져오기 (p. 109) 절차를 검토하여 인증서를 올바르게 업로드했는지 확인합니다.

     이 설정을 선택한 경우 대체 도메인 이름만 객체 URL에 사용하는 것이 좋습니다 (`https://example.com/logo.jpg`). CloudFront 배포 도메인 이름(`https://d111111abcdef8.cloudfront.net/logo.jpg`)을 사용하는 경우, Clients Supported(지원되는 클라이언트)에서 선택하는 값에 따라 뷰어가 다음과 같이 동작할 수 있습니다.

     - All Clients(모든 클라이언트): 뷰어에서 SNI가 지원되지 않는 경우 CloudFront 도메인 이름이 TLS/SSL 인증서의 도메인 이름과 일치하지 않으므로 경고가 표시됩니다.
     - Only Clients that Support Server Name Indication (SNI)(서버 이름 표시(SNI)를 지원하는 클라이언트만): CloudFront에서 객체를 반환하지 않고 해당 뷰어와의 연결을 끊습니다.

   클라이언트 지원(웹 배포만 해당)

   다음 옵션을 선택합니다.

- All Clients(모든 클라이언트): CloudFront가 전용 IP 주소를 사용하여 HTTPS 콘텐츠를 제공합니다. 이 옵션을 선택하면 SSL/TLS 인증서를 활성화된 배포와 연결할 때 추가 요금이 발생합니다. 자세한 내용은 http://aws.amazon.com/cloudfront/pricing을 참조하십시오.
- Only Clients that Support Server Name Indication (SNI)(서버 이름 표시(SNI)를 지원하는 클라이언트만): SNI를 지원하지 않는 이전 브라우저 또는 기타 클라이언트는 다른 방법을 사용하여 콘텐츠에 액세스해야 합니다.

자세한 내용은 CloudFront에서 HTTPS 요청을 제공하는 방식 선택 (p. 102) 단원을 참조하십시오.

5. 예, 편집합니다를 선택합니다.
6. 배포에 대한 일반 탭에서 Distribution Status(배포 상태)가 배포 완료로 변경되었는지 확인합니다. 배포에 대한 업데이트가 배포되기 전에 대체 도메인 이름을 사용하려고 하면 다음 단계에서 생성하는 링크가 작동하지 않을 수도 있습니다.
7. 도메인(예: example.com)에 대한 트래픽을 배포의 CloudFront 도메인 이름(예: d111111abcdef8.cloudfront.net)으로 라우팅하도록 도메인에 대한 DNS 서비스를 구성합니다. 사용하는 방법은 Route 53을 도메인의 DNS 서비스 공급자로 사용하는지, 다른 공급자로 사용하는지에 따라 결정됩니다.

   Note

   DNS 레코드가 업데이트 중인 배포가 아닌 다른 배포를 이미 가리키고 있는 경우, DNS를 업데이트하지 않고는 대체 도메인 이름을 배포에 추가할 수 없습니다. 자세한 내용은 대체 도메인 이름에 대한 제한 (p. 74) 단원을 참조하십시오.

   Route 53

   별칭 리소스 레코드 세트를 생성합니다. 별칭 리소스 레코드 세트를 사용하면 Route 53 쿼리에 대해서는 요금이 지불되지 않습니다. 또한 DNS에서 CNAME을 허용하지 않는 루트 도메인 이름(example.com)에 대해 별칭 리소스 레코드 세트를 생성할 수 있습니다. 자세한 내용은 Amazon Route 53 개발자 안내서에서 Amazon CloudFront 배포로 쿼리 라우팅 단원을 참조하십시오.

   다른 DNS 서비스 공급자

   DNS 서비스 공급자가 제공하는 방법을 사용하여 CNAME 리소스 레코드 세트를 도메인의 호스팅 영역에 추가합니다. 이 새 CNAME 리소스 레코드 세트는 DNS 쿼리를 도메인(예: www.example.com)에서 배포의 CloudFront 도메인 이름(예: d111111abcdef8.cloudfront.net)으로 리디렉션합니다. 자세한 내용은 DNS 서비스 공급자가 제공하는 설명서를 참조하십시오.

   Important

   도메인 이름에 대한 기존 CNAME 레코드가 이미 있는 경우 해당 리소스 레코드 세트를 업데이트하거나 배포의 CloudFront 도메인 이름을 가리키는 새 리소스 레코드 세트로 바꿉니다.
   또한, CNAME 리소스 레코드 세트가 오리진 서버의 도메인 이름이 아닌 배포의 도메인 이름을 가리키는지 확인합니다.
8. dig 또는 이와 유사한 도구를 사용하여 7단계에서 생성한 리소스 레코드 세트가 배포의 도메인 이름을 가리키는지 확인합니다. dig에 대한 자세한 내용은 http://www.kloth.net/services/dig.php를 참조하십시오.

   다음 예는 images.example.com 도메인에 대한 dig 요청과 응답의 관련 부분을 보여 줍니다.

```
[prompt]--> dig images.example.com

; <<> DiG 9.3.3rc2 <<> images.example.com
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15917
;; flags: qr rd ra; QUERY: 1, ANSWER: 9, AUTHORITY: 2, ADDITIONAL: 0

;; QUESTION SECTION:
```

```
;images.example.com.      IN    A

;; ANSWER SECTION:
images.example.com. 10800 IN CNAME d111111abcdef8.cloudfront.net.
...
...
```

Answer Section의 코드 행에서는 CNAME 리소스 레코드 세트가 images.example.com에 대한 쿼리를 CloudFront 배포 도메인 이름 d111111abcdef8.cloudfront.net으로 라우팅합니다. `CNAME`의 오른쪽에 있는 이름이 CloudFront 배포의 도메인 이름이면 CNAME 리소스 레코드 세트가 올바르게 구성된 것입니다. 이 값이 다른 값이면(예: Amazon S3 버킷의 도메인 이름) CNAME 리소스 레코드 세트가 잘못 구성된 것입니다. 이 경우에는 4단계로 돌아가서 배포의 도메인 이름을 가리키도록 CNAME 레코드를 수정합니다.

9. URL에서 배포의 CloudFront 도메인 이름 대신에 고유 도메인 이름을 사용하는 테스트 링크를 생성하여 대체 도메인 이름을 테스트합니다.
10. 애플리케이션에서 CloudFront 배포의 도메인 이름 대신에 대체 도메인 이름을 사용하도록 객체 링크를 변경합니다.

대체 도메인 이름을 다른 CloudFront 배포로 이동

CloudFront 배포 간에 대체 도메인 이름을 이동시키고 싶은 경우에 거쳐야 하는 단계들은 이동시키려는 도메인 이름에 따라 다릅니다.

- `marketing.example.com` 같은 하위 도메인의 경우에는 도메인 자체를 이동시킬 수 있습니다. 자세한 단계는 marketing.example.com 같은 하위 도메인 이름을 또 다른 배포로 이동시킵니다. (p. 69) 단원을 참조하십시오.
- `example.com`(2단계 도메인)과 같은 도메인의 경우에는 도메인을 또 다른 배포로 이동시키기 위해 AWS 지원을 처리해야 합니다(example.com 같은 도메인 이름을 또 다른 배포로 이동시킵니다. (p. 71)).

marketing.example.com 같은 하위 도메인 이름을 또 다른 배포로 이동시킵니다.

다음 단계를 따라 하위 도메인 이름을 이동시킵니다.

하위 도메인 이름을 새 배포로 이동시키려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 도메인 이름을 이동시킬 새 배포가 없는 경우에는 하나 생성합니다. 자세한 내용은 배포 만들기 (p. 35) 단원을 참조하십시오.
3. 해당 배포에 별칭 레코드 세트 또는 CNAME 레코드에 대한 와일드카드가 포함된 대체 도메인 이름을 추가합니다. 예를 들어 새 배포로 이동시키려는 하위 도메인 이름이 `marketing.example.com`인 경우에는 대체 도메인 이름 `*.example.com`을 추가합니다. 자세한 내용은 CloudFront에 추가하는 대체 도메인 이름에서 와일드카드 사용 (p. 73) 단원을 참조하십시오.

   Note

   `*.com`과 같은 최상위 도메인 이름에는 와일드카드를 추가할 수 없습니다. 따라서 `example.com`과 같은 도메인 이름을 새 배포로 이동시키고 싶은 경우에는 example.com 같은 도메인 이름을 또 다른 배포로 이동시킵니다. (p. 71) 단원을 참조하십시오.
4. 하위 도메인이 새 배포를 가리키도록 DNS 구성을 업데이트합니다. 예를 들어 하위 도메인 `marketing.example.com`이 배포에 대한 CloudFront 도메인 이름 `d111111abcdef8.cloudfront.net`으로 트래픽을 라우팅하도록 DNS 서비스를 업데이트할 수 있습니다.

구성을 업데이트하려면 다음 중 하나를 수행합니다.

- Route 53을 사용 중인 경우에는 원래 대체 도메인 이름을 설정한 방법에 따라 별칭 레코드나 CNAME 레코드를 업데이트합니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 레코드 편집 단원을 참조하십시오.
- 또 다른 DNS 서비스 공급자를 사용하고 있는 경우에는 DNS 서비스 공급자가 제공한 방법을 사용하여 CloudFront로 트래픽을 전달하는 CNAME 레코드를 업데이트합니다. 자세한 내용은 DNS 서비스 공급자가 제공하는 설명서를 참조하십시오.

  Note

  대체 도메인이 현재 구성되어 있으므로 이 시점에 하위 도메인은 원래 배포에서 계속 제공됩니다.

5. dig 또는 이와 유사한 도구를 사용하여 4단계에서 생성한 리소스 레코드 세트가 배포의 도메인 이름을 가리키는지 확인합니다. dig에 대한 자세한 내용은 http://www.kloth.net/services/dig.php를 참조하십시오.

   다음 예는 images.example.com 도메인에 대한 dig 요청과 응답의 관련 부분을 보여 줍니다.

```
[prompt]--> dig images.example.com

; <<> DiG 9.3.3rc2 <<> images.example.com
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15917
;; flags: qr rd ra; QUERY: 1, ANSWER: 9, AUTHORITY: 2, ADDITIONAL: 0

;; QUESTION SECTION:
;images.example.com.     IN    A

;; ANSWER SECTION:
images.example.com. 10800 IN CNAME d111111abcdef8.cloudfront.net.
...
...
```

   Answer Section의 코드 행에서는 CNAME 리소스 레코드 세트가 images.example.com에 대한 쿼리를 CloudFront 배포 도메인 이름 d111111abcdef8.cloudfront.net으로 라우팅합니다. `CNAME`의 오른쪽에 있는 이름이 CloudFront 배포의 도메인 이름이면 CNAME 리소스 레코드 세트가 올바르게 구성된 것입니다. 이 값이 다른 값이면(예: Amazon S3 버킷의 도메인 이름) CNAME 리소스 레코드 세트가 잘못 구성된 것입니다. 이 경우에는 4단계로 돌아가서 배포의 도메인 이름을 가리키도록 CNAME 레코드를 수정합니다.

   Note

   Route 53 별칭 레코드를 사용하는 경우, dig를 사용하여 리소스 레코드가 새 배포를 가리키는지 확인할 수 없습니다. 이 경우 리소스 레코드의 유형을 별칭에서 CNAME으로 변경하거나 레코드의 TTL(Time To Live) 값이 만료될 때까지 기다릴 수 있습니다.

6. 기존 배포에서 CNAME을 제거한 다음 이전에 와일드카드 대체 이름이 추가된 새 CloudFront 배포에 CNAME을 추가합니다.

   Note

   이러한 변경 내용을 전파하는 동안 대체 도메인 이름은 원래 배포 또는 새 배포에서 임의로 제공될 수 있습니다. 이러한 동작은 두 배포 경우 모두 배포 완료 상태에 도달한 후에도 몇 분 동안 지속될 수 있습니다.

7. URL에서 배포의 CloudFront 도메인 이름 대신에 고유 도메인 이름을 사용하는 테스트 링크를 생성하여 대체 도메인 이름을 테스트합니다.
8. 이제 와일드카드 대체 이름을 새 배포에서 제거할 수 있습니다.

9. 원래 배포를 더 이상 사용하지 않는 경우에는 삭제합니다. 자세한 내용은 배포 삭제 (p. 60) 단원을 참조하십시오.

example.com 같은 도메인 이름을 또 다른 배포로 이동시킵니다.

example.com 같은 2단계 도메인 이름의 경우, 도메인 이름을 또 다른 CloudFront 배포로 이동시키려면 AWS Support 팀에 문의해야 합니다. 앞에서 설명한 바와 같이 도메인을 직접 이동시키기 위해서는 도메인 이름의 일부로 와일드카드를 사용하여 도메인 라우팅을 설정해야 하기 때문에 추가적인 단계가 필요합니다. 2단계 도메인의 경우 이 단계에서 허용되지 않는 *.com으로 라우팅을 설정해야 합니다.

시작에 앞서 도메인 이름을 이동시킬 새 배포가 없는 경우에는 하나 생성합니다. 자세한 내용은 배포 만들기 (p. 35) 단원을 참조하십시오.

example.com 같은 도메인 이름을 새 배포로 이동시키는 절차는 다음과 같이 2단계로 구성됩니다.

1단계: DNS 서비스 공급자의 도메인에 대한 TXT 레코드를 생성하여 AWS 지원 팀에 도메인 이름을 소유하고 있다는 것을 증명합니다. 이렇게 하면 다른 사람이 배포 구성을 변경하지 못하도록 막을 수 있습니다.

2단계: AWS 지원 팀에게 새 CloudFront 배포로 도메인을 이동시켜달라고 요청합니다.

여기에는 구체적인 단계들이 나와 있습니다.

1단계: 도메인에 대한 TXT 레코드 생성

1. DNS 서비스 공급자 웹 사이트에 로그인합니다.

   서비스 공급자가 Route 53라면 Route 53 콘솔에 로그인합니다.
2. 다음과 같이 도메인에 대한 TXT 레코드를 생성합니다.

   <도메인 이름> TXT <CloudFront배포 이름>

   예: `example.com TXT d123.cloudfront.net`

   - DNS 서비스 공급자가 Route 53이면 세부 단계를 위해 3단계로 이동합니다.
   - 또 다른 DNS 서비스 공급자가 도메인을 호스팅한 경우에는 DNS 서비스 공급자의 설명서를 참조하십시오. 서비스 공급자에게 대신 TXT 레코드를 생성해달라고 요청해야 할 수도 있습니다.

     Tip

     도메인이 CNAME 레코드와 동일한 정보를 가질 수 있도록 서비스 공급자가 TXT 이름을 허용하지 않는 경우에는 앞에 밑줄(_)이 추가된 도메인 이름을 사용하는 TXT 레코드를 생성하는 방법을 고려하십시오. 예제는 지식 센터의 "CNAME 이미 존재" 오류 해결 기사를 참조하십시오.
3. DNS 서비스 공급자가 Route 53이면 다음 단계에 따라 TXT 레코드를 생성하여 도메인 소유권을 증명합니다.

   a. 호스팅 영역 페이지에서 레코드를 편집하고자 하는 호스팅 영역의 행을 두 번 클릭합니다.
   b. [Create Record Set]를 선택합니다.
   c. 다음 값을 입력합니다.

      - 이름: 새 CloudFront 배포로 이동시키려는 도메인 이름.
      - 유형: TXT
      - 별칭: 없음
      - TTL: 60초
      - 값: 이 도메인 이름을 추가하려는 CloudFront 배포의 이름(예: d123.cloudfront.net).
      - 라우팅 정책: 간단

d. Create를 선택합니다.

2단계: AWS 지원 팀에게 새 CloudFront 배포로 도메인을 이동시켜달라고 요청합니다.

- AWS에 로그인하고 AWS 지원 팀에 연락해서 도메인 소유권을 확인하고 새 CloudFront 배포로 도메인을 이동시켜달라고 요청합니다.

  Note

  도메인에 대해 생성한 TXT 레코드가 보이기 전에는 AWS 지원 팀이 도메인 소유권을 확인할 수 없습니다. DNS 서비스 공급자에서 생성한 레코드는 DNS 시스템을 통해 전파되는 데 시간이 다소 걸릴 수 있다는 점에 유의하십시오.(최대 며칠이 소요).

대체 도메인 이름 제거

도메인 또는 하위 도메인에서 CloudFront 배포로의 트래픽 라우팅을 중단시키고 싶으면 여기 나오는 단계에 따라 DNS 구성과 CloudFront 배포를 모두 업데이트합니다.

배포에서 대체 도메인 이름을 제거하고 DNS 구성을 업데이트하는 것이 중요합니다. 이렇게 하면 나중에 CloudFront 배포에 도메인 이름을 연결하고 싶은 경우에 문제가 발생하는 것을 막을 수 있습니다. 대체 도메인 이름이 이미 하나의 배포에 연결되어 있는 경우에는 또 다른 연결을 설정할 수 없습니다.

Note

또 다른 배포에 추가할 수 있도록 이 배포에서 대체 도메인 이름을 제거하고 싶으면 대체 도메인 이름을 다른 CloudFront 배포로 이동 (p. 69)의 단계를 따릅니다. 도메인을 제거하는 대신에 여기 나온 단계에 따라 또 다른 배포에 도메인을 추가한 경우에는 도메인이 새 배포에 연결되지 않는 기간이 존재합니다. 왜냐하면 CloudFront가 엣지 로케이션에 대한 업데이트로 전파되고 있기 때문입니다.

배포에서 대체 도메인 이름을 제거하려면

1. 먼저 도메인에 대한 인터넷 트래픽을 CloudFront 배포가 아닌 또 다른 리소스(예: Elastic Load Balancing 로드 밸런서)로 라우팅합니다. 또는 CloudFront로 트래픽을 라우팅하고 있는 DNS 레코드를 삭제할 수 있습니다.

   도메인의 DNS 서비스에 따라 다음 중 하나를 수행합니다.

   - Route 53을 사용 중인 경우에는 별칭 레코드나 CNAME 레코드를 업데이트 또는 삭제합니다. 자세한 내용은 레코드 편집 또는 레코드 삭제 단원을 참조하십시오.
   - 또 다른 DNS 서비스 공급자를 사용하고 있는 경우에는 DNS 서비스 공급자가 제공한 방법을 사용하여 CloudFront로 트래픽을 전달하는 CNAME 레코드를 업데이트 또는 삭제합니다. 자세한 내용은 DNS 서비스 공급자가 제공하는 설명서를 참조하십시오.
2. 도메인의 DNS 레코드를 업데이트한 후에는 변경 사항이 전파되어 DNS 해석기가 새 리소스로 트래픽을 라우팅할 때까지 기다립니다. 라우팅이 완료되면 URL에서 도메인을 사용하는 몇 가지 테스트 링크를 생성하여 확인할 수 있습니다.
3. 이제 AWS Management 콘솔에 로그인하여 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 연 다음, 다음과 같은 방법으로 도메인 이름을 제거할 수 있도록 CloudFront 배포를 업데이트합니다.

   a. 업데이트하려는 배포의 ID를 선택합니다.
   b. [General] 탭에서 [Edit]를 선택합니다.
   c. 대체 도메인 이름 사용(CNAME)에서 배포에서 더 이상 사용하고 싶지 않은 대체 도메인 이름 또는 도메인 이름을 제거합니다.
   d. 예, 편집합니다를 선택합니다.

CloudFront에 추가하는 대체 도메인 이름에서 와일드카드 사용

대체 도메인 이름을 추가할 경우 하위 도메인을 개별적으로 추가하는 대신에 도메인 이름 시작 부분에 와일드카드(*)를 사용할 수 있습니다. 예를 들어 대체 도메인 이름 `*.example.com`을 사용하면 `www.example.com`, `product-name.example.com` 및 `marketing.product-name.example.com` 등 example.com으로 끝나는 모든 도메인 이름을 객체 URL에서 사용할 수 있습니다. 객체의 이름은 도메인 이름과 상관없이 동일하며, 예를 들면 다음과 같습니다.

```
www.example.com/images/image.jpg

product-name.example.com/images/image.jpg

marketing.product-name.example.com/images/image.jpg
```

와일드카드가 포함된 대체 도메인 이름에 대해 다음 요구 사항을 따르십시오.

- 대체 도메인 이름은 별표와 마침표(`*.`)로 시작해야 합니다.
- `*domain.example.com`과 같이 와일드카드를 사용하여 하위 도메인 이름의 일부를 대체할 수 없습니다.
- `subdomain.*.example.com`과 같이 도메인 이름의 중간에 있는 하위 도메인은 바꿀 수 없습니다.
- 와일드카드를 사용하는 대체 도메인 이름을 포함하여 모든 대체 도메인 이름은 인증서의 주체 대체 이름(SAN)으로 보호되어야 합니다.

`*.example.com` 등의 와일드카드 대체 도메인 이름에는 둘 다 같은 CloudFront 배포에 있거나 같은 AWS 계정을 사용하여 생성된 배포에 있는 한 `example.com` 등의 다른 대체 도메인 이름이 포함될 수 있습니다.

대체 도메인 이름 사용과 관련된 요구 사항

CloudFront에 사용할 대체 도메인 이름을 추가할 때 다음과 같은 요구 사항이 있습니다.

대체 도메인 이름은 소문자여야 합니다.

모든 대체 도메인 이름(CNAME)이 유효하려면 소문자여야 합니다.

대체 도메인 이름은 유효한 SSL/TLS 인증서에 포함되어야 합니다.

CloudFront 배포에 사용할 대체 도메인 이름(CNAME)을 추가하려면 대체 도메인 이름이 포함된 신뢰할 수 있는 유효한 SSL/TLS 인증서를 배포에 연결해야 합니다. 그러면 도메인의 인증서에 대한 액세스 권한이 있는 사람만 해당 도메인과 관련된 CloudFront CNAME과 연결할 수 있습니다.

신뢰할 수 있는 인증서는 ACM 또는 다른 유효한 인증 기관(CA)에서 발급하는 인증서입니다. 자체 서명된 인증서는 사용할 수 없습니다. CloudFront는 Mozilla와 동일한 인증 기관을 지원합니다. 최신 목록은 Mozilla에 포함된 CA 인증서 목록을 참조하십시오.

연결한 인증서를 사용하여 와일드카드가 포함된 대체 도메인 이름 등의 대체 도메인 이름을 확인하기 위해 CloudFront는 인증서에서 주제 대체 이름(SAN)을 확인합니다. 추가하는 대체 도메인 이름은 SAN에 포함되어야 합니다.

> **Note**
>
> 한 번에 인증서 하나만 CloudFront 배포에 연결할 수 있습니다.

다음 중 하나를 수행하여 배포에 특정 대체 도메인 이름을 추가할 권한을 입증합니다.

- `product-name.example.com` 같은 각 도메인 이름의 개별 인증서 연결
- 도메인 이름의 시작 부분에 * 와일드카드가 포함된 인증서를 추가하여 인증서 하나로 여러 하위 도메인 포괄. 와일드카드를 지정할 때 CloudFront에서 여러 하위 도메인을 대체 도메인 이름으로 추가할 수 있습니다.

다음 예에서는 인증서 작업에서 도메인 이름에 와일드카드를 사용하여 CloudFront에서 특정 대체 도메인 이름을 추가할 수 있는 권한을 부여하는 보여 줍니다.

- `marketing.example.com`을 대체 도메인 이름으로 추가하려고 합니다. 인증서에서 `*.example.com` 도메인 이름을 나열합니다. CloudFront에 이 인증서를 연결할 때 해당 수준에서 와일드카드를 대체하는 배포의 대체 도메인 이름을 추가할 수 있습니다(예: `marketing.example.com`). 또한 예를 들어 다음과 같은 대체 도메인 이름을 추가할 수도 있습니다.
  - `product.example.com`
  - `api.example.com`

  그러나 와일드카드보다 높거나 낮은 수준의 대체 도메인 이름은 추가할 수 없습ㄴ디ㅏ. 예를 들어 `example.com` 또는 `marketing.product.example.com` 대체 도메인 이름은 추가할 수 없습니다.
- `example.com`을 대체 도메인 이름으로 추가하려고 합니다. 이를 수행하려면 배포에 연결하는 인증서에서 도메인 이름 `example.com` 자체를 나열해야 합니다.
- `marketing.product.example.com`을 대체 도메인 이름으로 추가하려고 합니다. 이를 수행하려면 인증서에서 `*.product.example.com`을 나열하거나, 인증서에 `marketing.product.example.com` 자체를 나열하면 됩니다.

DNS 구성 변경 권한

대체 도메인 이름을 추가하는 경우, CNAME 레코드를 생성하여 도메인 이름에 대한 DNS 쿼리를 CloudFront 배포에 라우팅해야 합니다. 이를 수행하려면 사용 중인 대체 도메인 이름의 DNS 서비스 공급자를 통해 CNAME 레코드를 생성할 수 있는 권한이 있어야 합니다. 일반적으로 이 경우 도메인을 소유하고 있지만 도메인 소유자를 위해 애플리케이션을 개발하는 중일 수도 있습니다.

대체 도메인 이름과 HTTPS

최종 사용자가 대체 도메인 이름과 함께 HTTPS를 사용하도록 하려면 추가 구성을 수행해야 합니다. 자세한 내용은 대체 도메인 이름과 HTTPS 사용 (p. 102) 단원을 참조하십시오.

대체 도메인 이름에 대한 제한

대체 도메인 이름 사용에 대한 제한 사항은 다음과 같습니다.

최대 대체 도메인 이름 수

현재 하나의 배포에 추가할 수 있는 대체 도메인 이름의 최대 수를 살펴보고 더 높은 할당량(이전에는 제한이라고 함)을 요청하려면, 웹 배포의 일반 할당량 (p. 447) 단원을 참조하십시오.

중복 및 중첩 대체 도메인 이름

AWS 계정에서 다른 배포를 소유하더라도 대체 도메인 이름이 다른 CloudFront 배포 안에 이미 있는 경우에는 대체 도메인 이름을 CloudFront 배포에 추가할 수 없습니다.

하지만 `*.example.com` 등의 와일드카드 대체 도메인 이름을 추가할 수 있는데 여기에는 `www.example.com` 등의 비 와일드카드 대체 도메인 이름이 포함(중첩)됩니다. 중첩되는 대체 도메인 이름은 동일한 배포에 있거나, 두 배포가 모두 같은 AWS 계정을 사용하여 생성되는 한 별도의 배포에 있을 수 있습니다.

배포 2개의 대체 도메인 이름이 겹치는 경우 CloudFront는 DNS 레코드가 가리키는 배포에 상관 없이 보다 구체적으로 일치하는 이름을 사용하여 배포에 요청을 보냅니다. 예를 들어, `marketing.domain.com`은 `*.domain.com`보다 더 구체적입니다.

이미 배포를 가리키는 대체 도메인 이름

DNS 레코드가 생성 또는 수정 중인 배포가 아닌 다른 배포를 이미 가리키고 있는 경우, 다른 도메인 이름을 배포에 추가할 수 없습니다. 이 시나리오에서 CloudFront 배포의 도메인 이름을 추가하려면 먼저 DNS 공급자에서 DNS를 업데이트해야 합니다.

이 문제를 해결하려면 DNS 공급자에 로그인하여 기존 DNS 레코드를 제거하거나 DNS 공급자에게 문의하여 해당 레코드를 제거하도록 요청합니다. 그런 다음 배포의 대체 도메인 이름을 추가하거나 변경하

기 위한 단계를 따라 배포에 대한 올바른 DNS 레코드를 생성합니다. 자세한 내용은 대체 도메인 이름 추가 (p. 67) 또는 대체 도메인 이름을 다른 CloudFront 배포로 이동 (p. 69) 단원을 참조하십시오.

Domain Fronting(도메인 프론팅)

CloudFront에는 여러 AWS 계정에서 발생하는 도메인 프론팅에 대한 보호가 포함되어 있습니다. 표준이 아닌 클라이언트가 한 AWS 계정의 도메인 이름에 대한 TLS/SSL 연결을 생성하지만 다른 AWS 계정에서 관련이 없는 이름에 대한 HTTPS 요청을 하는 시나리오가 있습니다. 예를 들어, TLS 연결이 `www.example.com`에 연결된 후, `www.example.org`에 대한 요청을 발행할 수 있습니다.

도메인 프론팅이 여러 AWS 계정을 교차하는 경우를 방지하기 위해 CloudFront는 특정 연결을 위해 사용되는 인증서를 소유한 AWS 계정이 항상 동일한 연결에서 처리하는 요청을 소유한 AWS 계정과 일치하는지 확인합니다.

두 AWS 계정 번호가 일치하지 않으면 CloudFront는 421 Misdirected Request로 응답하여 올바른 도메인을 사용하여 연결할 기회를 클라이언트에 제공합니다.

도메인의 최상위 노드(Zone Apex)에서 대체 도메인 이름 추가

배포에 대체 도메인 이름을 추가하는 경우, 일반적으로 DNS 구성에서 CNAME 레코드를 생성하여 도메인 이름에 대한 DNS 쿼리를 CloudFront 배포에 라우팅합니다. 하지만 Zone Apex라고 하는 DNS 네임스페이스의 최상위 노드에 대한 CNAME 레코드를 생성할 수 없습니다. DNS 프로토콜이 허용하지 않습니다. 예를 들어 DNS 이름 `example.com`을 등록할 경우 zone apex는 `example.com`입니다. `example.com`에 대한 CNAME을 생성할 수는 있지만, `www.example.com`, `newproduct.example.com` 등에 대한 CNAME 레코드는 생성할 수 없습니다.

Route 53을 DNS 서비스로 사용하는 경우 CNAME 레코드보다 두 가지 장점이 있는 별칭 리소스 레코드 세트를 생성할 수 있습니다. 최상위 노드(example.com)에서 도메인 이름에 대한 별칭 리소스 레코드 세트를 생성할 수 있습니다. 또한 별칭 리소스 레코드 세트를 사용할 때는 Route 53 쿼리에 대한 요금을 지불하지 않습니다.

Note

IPv6를 활성화하는 경우 두 개의 별칭 리소스 레코드를 생성해야 합니다. 하나는 IPv6 트래픽(A 레코드)을 라우팅하고 다른 하나는 IPv4 트래픽(AAAA 레코드)을 라우팅합니다. 자세한 내용은 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 주제에서 IPv6 사용 (p. 55) 단원을 참조하십시오.

자세한 내용은 Amazon Route 53 개발자 안내서에서 Amazon CloudFront 배포로 쿼리 라우팅 단원을 참조하십시오.

CloudFront 배포를 통한 WebSocket 사용

Amazon CloudFront는 클라이언트와 서버 사이에 수명이 긴 양방향 연결이 필요할 때 유용한 TCP 기반 프로토콜인 WebSocket의 사용을 지원합니다. 지속적 연결이 실시간 애플리케이션의 요구 사항에 포함되는 경우가 흔히 있습니다. Websocket을 사용할 수 있는 시나리오에는 채팅 플랫폼, 온라인 협업 공간, 멀티 플레이어 게임 및 금융 거래 플랫폼과 같이 실시간 데이터 피드를 제공하는 서비스 등이 포함됩니다. WebSocket 연결에서는 데이터가 전이중 통신의 양방향으로 흐를 수 있습니다.

CloudFront는 별도의 추가적 구성 없이 WebSocket 연결을 전역적으로 지원합니다. 모든 CloudFront 배포에는, 클라이언트와 서버 모두 프로토콜을 지원하는 한, 내장된 WebSocket 프로토콜이 지원됩니다.

WebSocket 프로토콜의 작동 방식

WebSocket 프로토콜은 독립적인 TCP 기반의 프로토콜이며, 이를 통해 HTTP의 오버헤드 및 잠재적인 지연 시간 연장을 방지할 수 있습니다.

WebSocket 연결을 설정하기 위해, 클라이언트가 HTTP의 업그레이드 의미론을 사용하는 정규 HTTP 요청을 전송하여 프로토콜을 변경합니다. 그 후에는 서버가 핸드셰이크를 완료할 수 있습니다. WebSocket 연결

은 개방 상태를 유지하며 클라이언트 또는 서버가 매회 새로운 연결을 설정할 필요 없이 서로에게 데이터 프레임을 전송할 수 있습니다.

기본적으로 WebSocket 프로토콜은 정규 WebSocket 연결에 포트 80을, TLS/SSL을 통한 WebSocket 연결에는 포트 443을 사용합니다. CloudFront Viewer Protocol Policy (p. 45)와 오리진 프로토콜 정책 (p. 41)데 대해 선택한 옵션은 HTTP 트래픽뿐 아니라 WebSocket 연결에도 적용됩니다.

WebSocket 요구 사항

WebSocket 요청은 다음 표준 형식의 RFC 6455(http://tools.ietf.org/html/rfc6455)를 준수해야 합니다.

예제 클라이언트 요청:

```
GET /chat HTTP/1.1
        Host: server.example.com
        Upgrade: websocket
        Connection: Upgrade
        Sec-WebSocket-Key: dGhlIHNhbXBsZSBub25jZQ==
        Origin: http://example.com
        Sec-WebSocket-Protocol: chat, superchat
        Sec-WebSocket-Version: 13
```

샘플 서버 응답:

```
HTTP/1.1 101 Switching Protocols
        Upgrade: websocket
        Connection: Upgrade
        Sec-WebSocket-Accept: s3pPLMBiTxaQ9kYGzzhZRbK+xOo=
        Sec-WebSocket-Protocol: chat
```

WebSocket 연결이 클라이언트나 서버 또는 네트워크 장애로 인해 끊어진 겨우, 클라이언트 애플리케이션이 서버와의 연결을 다시 시작해야 합니다.

CloudFront가 배포하는 콘텐츠 추가, 제거 또는 바꾸기

이 단원에서는 최종 사용자에게 제공하려는 콘텐츠를 CloudFront가 액세스할 수 있게 하는 방법, 웹 사이트 또는 애플리케이션에서 객체를 지정하는 방법, 콘텐츠를 제거하거나 바꾸는 방법을 대해 설명합니다.

주제

- CloudFront가 배포하는 콘텐츠 추가 및 액세스 (p. 77)
- CloudFront 배포를 사용하여 기존 콘텐츠 업데이트 (p. 77)
- CloudFront가 콘텐츠를 배포하지 않도록 콘텐츠 제거 (p. 78)
- CloudFront에서 파일에 대한 URL 형식 사용자 지정 (p. 79)
- 파일 무효화 (p. 80)
- 압축된 파일 제공 (p. 87)

CloudFront가 배포하는 콘텐츠 추가 및 액세스

CloudFront가 콘텐츠(객체)를 배포하게 하려면, 배포를 위해 지정한 오리진 중 하나에 파일을 추가하고 파일에 대한 CloudFront 링크를 공개합니다. CloudFront 엣지 로케이션은 파일에 대한 최종 사용자 요청을 받기 전까지는 오리진으로부터 새 파일을 가져오지 않습니다. 자세한 내용은 CloudFront에서 콘텐츠를 제공하는 방법 (p. 4) 단원을 참조하십시오.

CloudFront에서 배포하려는 파일을 추가하는 경우, 배포에 지정된 Amazon S3 버킷 중 하나에 이를 추가해야 하고 사용자 오리진의 경우 지정된 도메인의 디렉터리에 이를 추가해야 합니다. 또한 해당 캐시 동작의 경로 패턴에서 정확한 오리진에 요청을 전송하는지 확인합니다.

예를 들어, 캐시 동작의 경로 패턴이 `*.html`이라고 가정합니다. 이 오리진으로 요청을 전송하도록 구성된 다른 캐시 동작이 없는 경우 CloudFront는 `*.html` 파일만 전송합니다. 예를 들어, 이 시나리오에서 CloudFront는 오리진에 업로드한 .jpg 파일을 배포하지 않습니다. .jpg 파일을 포함하는 캐시 동작을 생성하지 않았기 때문입니다.

CloudFront 서버는 제공하는 객체의 MIME 형식을 확인하지 않습니다. 오리진에 파일을 업로드할 경우 이 파일에 대한 `Content-Type` 헤더 필드를 설정하는 것이 좋습니다.

CloudFront 배포를 사용하여 기존 콘텐츠 업데이트

CloudFront가 배포하도록 설정된 기존 콘텐츠를 업데이트하는 방법에는 두 가지가 있습니다.

- 동일한 이름을 사용하여 파일 업데이트
- 파일 이름에 버전 식별자를 사용하여 업데이트

CloudFront가 제공하는 콘텐츠 관리에 대한 제어를 강화하기 위해 파일 이름이나 폴더 이름에 버전 식별자를 사용하는 것이 좋습니다.

버전이 지정된 파일 이름을 사용하여 기존 파일 업데이트

CloudFront 배포의 기존 파일을 업데이트할 경우, 콘텐츠를 보다 잘 제어할 수 있도록 파일 이름 또는 디렉터리 이름에 일종의 버전 식별자를 포함하는 것이 좋습니다. 이 식별자는 날짜-타임스탬프, 일련 번호 또는 동일 객체의 두 버전을 식별하는 기타 방법이 될 수 있습니다.

예를 들어, 그래픽 파일의 이름을 image.jpg라고 지정하는 대신 image_1.jpg라고 할 수 있습니다. 이 파일의 새 버전 제공을 시작하려는 경우, 새 파일의 이름을 image_2.jpg라고 지정하고 image_2.jpg 파일을 가리키도록 웹 애플리케이션 또는 웹 사이트의 링크를 업데이트할 수 있습니다. 또는, images_v1 디렉터리에 모든 그래픽 파일을 넣을 수 있으며 그래픽 파일 하나 이상의 새 버전 제공을 시작하려는 경우 새 images_v2 디렉터리를 만들고 이 디렉터리를 가리키도록 링크를 업데이트할 수 있습니다. 버전 관리를 사용하면 CloudFront에서 새 버전의 객체 제공을 시작하기 전에 객체가 만료될 때까지 기다릴 필요가 없으며, 객체 무효화에 비용을 지불할 필요도 없습니다.

파일의 버전을 관리하는 경우라도 만료 날짜는 설정하는 것이 좋습니다. 자세한 내용은 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204) 단원을 참조하십시오.

> **Note**
>
> 버전이 지정된 파일 이름 또는 디렉터리 이름을 지정하는 것은 Amazon S3 객체 버전 관리와는 관련이 없습니다.

동일한 파일 이름을 사용하여 기존 콘텐츠 업데이트

CloudFront 배포의 기존 파일을 업데이트하고, 동일한 파일 이름을 사용할 수 있긴 하지만 권장되는 방법은 아닙니다. CloudFront에서는 오리진에 새 파일 또는 업데이트된 파일을 저장하는 경우가 아니라 파일이 요청되는 경우에만 엣지 로케이션에 파일을 배포합니다. 이름이 같은 최신 버전으로 오리진에서 기존 파일을 업데이트한 경우, 다음 두 가지 상황이 모두 발생하기 전까지는 엣지 로케이션에서 오리진으로부터 새 버전을 가져오지 않습니다.

- 캐시에 있는 파일의 기존 버전은 만료됩니다. 자세한 내용은 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204) 단원을 참조하십시오.
- 엣지 로케이션에는 파일에 대한 사용자 요청이 존재합니다.

파일을 바꿀 때 동일한 이름을 사용하면 CloudFront에서 새 파일 제공을 시작하는 시기를 제어할 수 없습니다. CloudFront는 기본적으로 24시간 동안 엣지 로케이션에 파일을 캐싱합니다. (자세한 내용은 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204) 단원을 참조하십시오.) 예를 들어, 다음과 같이 전체 웹 사이트의 모든 파일을 바꾸려는 경우

- 사용 빈도가 낮은 페이지의 파일은 엣지 로케이션에 없을 수 있습니다. 이러한 파일의 새 버전은 다음 요청 시에 제공을 시작합니다.
- 일부 페이지의 파일은 어떤 엣지 로케이션에는 있고 다른 엣지 로케이션에는 없을 수 있습니다. 따라서 최종 사용자에게는 제공되는 엣지 로케이션에 따라 서로 다른 버전의 객체가 표시될 수 있습니다.
- 사용 빈도가 높은 페이지의 경우 파일을 새 버전으로 바꾸기 직전에 CloudFront가 해당 페이지의 파일을 가져왔을 수 있으므로 새 버전의 파일이 최대 24시간 동안 제공되지 않을 수 있습니다.

CloudFront가 콘텐츠를 배포하지 않도록 콘텐츠 제거

CloudFront 배포에 더 이상 포함시키지 않으려는 오리진에서 파일을 제거할 수 있습니다. 하지만 CloudFront는 파일이 만료될 때까지 엣지 캐시의 콘텐츠를 최종 사용자에게 계속 표시합니다.

파일을 즉시 제거하려면 다음 방법 중 하나를 사용해야 합니다.

- 파일을 무효화합니다. 자세한 내용은 파일 무효화 (p. 80) 단원을 참조하십시오.
- 파일 버전 관리를 사용합니다. 버전 관리를 사용하는 경우 한 파일의 여러 버전은 서로 다른 이름을 사용합니다. CloudFront 배포에 이 이름을 사용하여 최종 사용자에게 반환되는 파일을 변경할 수 있습니다. 자세한 내용은 버전이 지정된 파일 이름을 사용하여 기존 파일 업데이트 (p. 78) 단원을 참조하십시오.

CloudFront에서 파일에 대한 URL 형식 사용자 지정

CloudFront가 최종 사용자에게 제공하게 하려는 객체(콘텐츠)와 함께 오리진을 설정한 후, 올바른 URL을 사용하여 웹 사이트 또는 애플리케이션 코드에 해당 객체를 참조해야 CloudFront가 객체를 제공할 수 있습니다.

웹 페이지 또는 웹 애플리케이션에서 객체의 URL에 사용하는 도메인 이름은 다음 중 하나일 수 있습니다.

- 배포를 생성할 때 CloudFront에서 자동으로 할당하는 도메인 이름(예: `d111111abcdef8.cloudfront.net`)
- 고유의 도메인 이름(예: `example.com`)

예를 들어, 다음 URL 중 하나를 사용하여 `image.jpg` 파일을 반환할 수 있습니다.

```
http://d111111abcdef8.cloudfront.net/images/image.jpg
```

```
http://example.com/images/image.jpg
```

콘텐츠를 Amazon S3 버킷에 저장하든지 또는 사용자 지정 오리진(예: 자체 웹 서버 중 하나)에 저장하든지 상관없이 동일한 URL 형식을 사용합니다.

> **Note**
>
> URL 형식은 배포에서 Origin Path(오리진 경로)에 지정하는 값에 따라 부분적으로 달라집니다. 이 값은 CloudFront에 객체의 최상위 디렉터리 경로를 제공합니다. 웹 배포를 생성할 때 오리진 경로를 설정하는 방법에 대한 자세한 내용은 Origin Path (p. 39) 단원을 참조하십시오.

URL 형식에 대한 자세한 내용은 다음 단원을 참조하십시오.

고유의 도메인 이름 사용(Example.com)

배포를 생성할 때 CloudFront에서 할당하는 기본 도메인 이름을 사용하는 대신, `example.com`과 같은 대체 도메인 이름을 추가할 수 있습니다. CloudFront를 사용하여 고유의 도메인 이름을 설정하면 배포에서 객체에 이와 같은 URL을 사용할 수 있습니다.

```
http://example.com/images/image.jpg
```

최종 사용자와 CloudFront 간에 HTTPS를 사용하려는 경우 대체 도메인 이름과 HTTPS 사용 (p. 102) 단원을 참조하십시오.

URL에서 후행 슬래시(/) 사용

CloudFront 배포에서 디렉터리에 대한 URL을 지정할 때 후행 슬래시를 항상 사용하거나 후행 슬래시를 사용하지 않도록 선택합니다. 예를 들어, 모든 URL에 대해 다음 형식 중 하나만 선택합니다.

```
http://d111111abcdef8.cloudfront.net/images/
```

```
http://d111111abcdef8.cloudfront.net/images
```

이 선택이 중요한 이유

두 형식 모두 CloudFront 객체에 연결할 수 있지만, 형식을 일치시키면 나중에 디렉터리를 무효화하려는 경우 문제를 방지할 수 있습니다. CloudFront는 후행 슬래시를 포함하여 지정된 그대로 URL을 저장합니다. 따라서 형식이 일치하지 않는 경우 CloudFront에서 디렉터리가 확실히 제거되도록 슬래시가 있는 디렉터리 URL과 슬래시가 없는 디렉터리 URL을 모두 무효화해야 합니다.

두 URL 형식을 모두 무효화하는 작업은 불편하며 추가 비용이 발생할 수 있습니다. 두 가지 유형의 URL을 모두 처리하기 위해 무효화를 중복해서 수행해야 하는 경우 해당 월에 허용되는 무료 무효화의 최대 수를 초과할 수 있기 때문입니다. 그리고 이러한 경우 CloudFront에 각 디렉터리 URL의 형식이 하나만 있더라도 모든 무효화 요금을 지불해야 합니다.

제한된 콘텐츠에 대한 서명된 URL 생성

액세스를 제한하려는 콘텐츠가 있는 경우 서명된 URL을 생성할 수 있습니다. 예를 들어, 인증한 사용자에게만 콘텐츠를 배포하려는 경우 지정된 기간 동안에만 유효하거나 지정된 IP 주소에서만 사용할 수 있는 URL을 생성할 수 있습니다. 자세한 내용은 서명된 URL과 서명된 쿠키를 사용하여 프라이빗 콘텐츠 제공 (p. 115) 단원을 참조하십시오.

RTMP 배포 미디어 파일에 대한 URL 생성

RTMP 배포를 사용하여 스트리밍 콘텐츠를 제공하는 경우, 파일의 URL에 추가 문자를 포함시켜야 합니다. 자세한 내용은 미디어 플레이어 구성 (p. 277) 단원을 참조하십시오.

파일 무효화

CloudFront 엣지 캐시에서 파일이 만료되기 전에 파일을 제거해야 할 경우, 다음 방법 중 하나를 사용하면 됩니다.

- 엣지 캐시에서 파일을 무효화합니다. 다음에 최종 사용자가 파일을 요청하면 CloudFront는 최신 버전의 파일을 가져오도록 오리진에 반환합니다.
- 파일 버전 관리를 사용하여 서로 다른 이름을 가진 여러 버전의 파일을 제공합니다. 자세한 내용은 버전이 지정된 파일 이름을 사용하여 기존 파일 업데이트 (p. 78) 단원을 참조하십시오.

> **Important**
>
> RTMP 배포로 제공되는 객체는 무효화할 수 없습니다.

파일을 무효화하려면 개별 파일에 대한 경로나 `*` 와일드카드로 끝나는 경로 중에 지정할 수 있습니다. 이는 다음 예에 나와 있듯 각각 하나의 파일 또는 여러 파일에 적용됩니다.

- `/images/image1.jpg`
- `/images/image*`
- `/images/*`

> **Note**
>
> AWS 명령줄 인터페이스(CLI)를 사용하여 파일을 무효화하고 `*` 와일드카드를 포함하는 경로를 지정하는 경우에는 경로를 따옴표(")로 묶어야 합니다.
> 예: `aws cloudfront create-invalidation --distribution-id $CDN_DISTRIBUTION_ID --paths "/*"`

매월 지정된 개수의 무효화 경로를 무료로 제출할 수 있습니다. 한 달에 할당된 개수의 무효화 경로를 초과하여 제출하는 경우, 제출한 무효화 경로 각각에 대해 요금을 지불합니다. 무효화에 부과되는 요금에 대한 자세한 내용은 파일 무효화에 대한 요금 결제 (p. 86) 단원을 참조하십시오.

주제

- 파일을 무효화하는 방법과 버전 지정된 파일 이름을 사용하는 방법 중에서 선택 (p. 81)
- 무효화할 파일 결정 (p. 81)
- 무효화할 파일 지정 (p. 81)
- 콘솔을 사용하여 파일 무효화 (p. 84)
- CloudFront API를 사용하여 파일 무효화 (p. 86)
- 파일 무효화를 위한 타사 도구 (p. 86)
- 동시 무효화 요청 최대값 (p. 86)
- 파일 무효화에 대한 요금 결제 (p. 86)

파일을 무효화하는 방법과 버전 지정된 파일 이름을 사용하는 방법 중에서 선택

배포가 제공하는 파일의 버전을 제어하는 방법에는 파일을 무효화하는 방법과, 버전 지정된 파일 이름을 제공하는 방법이 있습니다. 파일을 자주 업데이트하려는 경우, 기본적으로 파일 버전 관리를 사용하는 것이 좋습니다. 그 이유는 다음과 같습니다.

- 버전 관리를 사용하면 사용자가 로컬에 또는 회사 캐싱 프록시 뒤에 캐싱된 버전을 가지고 있더라도 요청이 반환하는 파일을 제어할 수 있습니다. 파일을 무효화하면 파일이 해당 캐시에서 만료될 때까지는 사용자에게 이전 버전이 계속 표시될 수 있습니다.
- 버전 관리를 통해 파일 변경의 결과를 더 쉽게 분석할 수 있도록 CloudFront 액세스 로그에는 파일의 이름이 포함됩니다.
- 버전 관리를 사용하면 서로 다른 사용자에게 여러 버전의 파일을 제공할 수 있습니다.
- 버전 관리를 통해 파일 개정 버전을 손쉽게 앞 버전이나 뒤 버전으로 이동할 수 있습니다.
- 버전 관리는 비용이 저렴합니다. CloudFront가 새 버전의 파일을 엣지 로케이션에 전송하게 하려면 요금이 발생하지만, 파일 무효화에 대해서는 요금을 지불할 필요가 없습니다.

파일 버전 관리에 대한 자세한 내용은 버전이 지정된 파일 이름을 사용하여 기존 파일 업데이트 (p. 78) 단원을 참조하십시오.

무효화할 파일 결정

특정 디렉터리에 있는 모든 파일이나 같은 문자로 시작하는 이름을 가진 모든 파일 등 여러 파일을 무효화하려는 경우, 무효화 경로의 끝에 * 와일드카드를 포함시키면 됩니다. * 와일드카드 사용에 대한 자세한 내용은 무효화 경로 단원을 참조하십시오.

선택한 파일을 무효화하려 하지만 사용자가 오리진의 모든 파일에 반드시 액세스할 필요가 없는 경우, CloudFront에서 최종 사용자가 요청한 파일을 확인하여 그러한 파일만 무효화할 수 있습니다. 최종 사용자가 요청한 파일을 확인하려면 CloudFront 액세스 로깅을 활성화합니다. 액세스 로그에 대한 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

무효화할 파일 지정

파일을 무효화하기 위해 CloudFront 콘솔을 사용하든 또는 CloudFront API를 사용하든 상관없이 파일에 지정하는 요구 사항과 제한은 동일합니다. 무효화하려는 파일을 지정할 때는 다음 사항에 유의하십시오.

대소문자 구분

무효화 경로는 대소문자를 구분하므로 `/images/image.jpg`와 `/images/Image.jpg`는 서로 다른 두 파일을 지정합니다.

Lambda 함수를 사용하여 URI 변경

CloudFront 배포가 최종 사용자 요청 이벤트에서 Lambda 함수를 트리거하고 해당 함수가 요청된 파일의 URI를 변경하는 경우 CloudFront 엣지 캐시에서 파일을 제거하려면 두 URI를 모두 무효화하는 것이 좋습니다.

- 최종 사용자 요청의 URI
- 함수가 변경한 후의 URI

예를 들어, Lambda 함수가 파일의 URI를 다음에서

```
http://d111111abcdef8.cloudfront.net/index.html
```

언어 디렉터리가 포함된 다음 URI로 변경한다고 가정합니다.

```
http://d111111abcdef8.cloudfront.net/en/index.html
```

파일을 무효화하려면 다음 경로를 지정해야 합니다.

- `index.html`
- `en/index.html`

자세한 내용은 무효화 경로 단원을 참조하십시오.

기본 루트 객체

기본 루트 객체(파일)를 무효화하려면 다른 파일에 대한 경로를 지정할 때와 같은 방법으로 경로를 지정합니다.

배포 유형

웹 배포와 연결된 파일만 무효화할 수 있습니다.

쿠키 전달

오리진에 쿠키를 전달하도록 CloudFront를 구성하는 경우, CloudFront 엣지 캐시에는 여러 버전의 파일이 포함되어 있을 수 있습니다. 파일을 무효화할 경우 CloudFront에서는 연결된 쿠키와 상관없이 각 캐싱된 버전의 파일을 무효화합니다. 연결된 쿠키에 따라 일부 버전만 선택적으로 무효화하고 나머지는 무효화하지 않을 수는 없습니다. 자세한 내용은 쿠키를 기반으로 콘텐츠 캐싱 (p. 198) 단원을 참조하십시오.

헤더 전달

오리진에 헤더의 화이트리스트를 전달하고 헤더 값에 따라 캐싱하도록 CloudFront를 구성한 경우, CloudFront 엣지 캐시에는 여러 버전의 파일이 포함되어 있을 수 있습니다. 파일을 무효화할 경우 CloudFront에서는 헤더 값과 상관없이 각 캐싱된 버전의 파일을 무효화합니다. 헤더 값에 따라 일부 버전만 선택적으로 무효화하고 나머지는 무효화하지 않을 수는 없습니다. 전체 헤더를 오리진에 전달하도록 CloudFront를 구성한 경우, CloudFront에서는 파일을 캐싱하지 않습니다. 자세한 내용은 요청 헤더에 따라 콘텐츠 캐싱 (p. 200) 단원을 참조하십시오.

쿼리 문자열 전달

CloudFront에서 쿼리 문자열을 오리진에 전달하도록 구성한 경우, 다음 예제에서와 같이 파일 무효화 시에는 쿼리 문자열을 포함해야 합니다.

- `images/image.jpg?parameter1=a`
- `images/image.jpg?parameter1=b`

클라이언트 요청에 동일 파일에 대한 5가지 쿼리 문자열을 포함하는 경우, 파일을 5번 무효화하거나 각 쿼리 문자열에 대해 한 번만 무효화할 수 있습니다. 또는 다음 예제에서와 같이 * 와일드카드를 무효화 경로에 사용할 수 있습니다.

```
/images/image.jpg*
```

무효화 경로에 와일드카드를 사용하는 방법에 대한 자세한 내용은 무효화 경로 단원을 참조하십시오. 쿼리 문자열에 대한 자세한 내용은 쿼리 문자열 파라미터 기반의 콘텐츠 캐싱 (p. 196) 단원을 참조하십시오. 사용 중인 쿼리 문자열을 확인하기 위해서는 CloudFront 로깅을 활성화하면 됩니다. 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

허용되는 최대값

허용되는 최대 무효화 수에 대한 자세한 내용은 동시 무효화 요청 최대값 (p. 86) 단원을 참조하십시오.

Microsoft Smooth Streaming 파일

해당 캐시 동작에 대해 Smooth Streaming을 활성화한 경우 Microsoft Smooth Streaming 포맷의 미디어 파일을 무효화할 수 없습니다.

경로의 비 ASCII 또는 안전하지 않은 문자

경로에 비 ASCII 문자나 RFC 1783(http://www.ietf.org/rfc/rfc1738.txt)에 정의된 안전하지 않은 문자가 포함된 경우, 그러한 문자를 URL로 인코딩합니다. 경로에 있는 그 밖의 다른 문자는 URL로 인코딩하지 마세요. 그렇지 않으면 CloudFront에서 기존 버전의 업데이트된 파일을 무효화하지 않습니다.

무효화 경로

이 경로는 배포에 상대적입니다. 선행 /(슬래시) 사용은 선택 사항입니다. 예를 들어, `http://d111111abcdef8.cloudfront.net/images/image2.jpg`의 파일을 무효화하려면 다음과 같이 지정할 수 있습니다.

```
/images/image2.jpg
```

또는

```
images/image2.jpg
```

또한 `*` 와일드카드를 사용하여 여러 파일을 동시에 무효화할 수도 있습니다. `*`는 0개 이상의 문자로 대체되며 무효화 경로의 마지막 문자여야 합니다. AWS 명령줄 인터페이스(CLI)를 사용하여 파일을 무효화하고 `*` 와일드 카드를 포함하는 경로를 지정하는 경우에는 경로 주변에 따옴표(`"`)를 사용해야 합니다(예시: `"/*"`).

다음은 몇 가지 예입니다:

- 특정 디렉터리의 모든 파일을 무효화하려면

  ```
  /directory-path/*
  ```

- 특정 디렉터리와 그 하위 디렉터리 전체, 디렉터리와 하위 디렉터리에 들어 있는 모든 파일을 무효화하려면

  ```
  /directory-path*
  ```

- logo.jpg, logo.png, log.gif와 같이 같은 이름에 다른 파일 이름 확장명을 지닌 모든 파일을 무효화하려면

  ```
  /directory-path/file-name.*
  ```

- 특정 디렉터리의 전체 파일 중 파일 이름이 같은 문자로 시작하는 파일(예: HLS 포맷 비디오용 파일 전체)을 무효화하려면

  ```
  /directory-path/initial-characters-in-file-name*
  ```

- 쿼리 문자열 파라미터에 따라 캐싱하도록 CloudFront를 구성할 때 각 버전의 파일을 무효화하려는 경우

  ```
  /directory-path/file-name.file-name-extension*
  ```

- 특정 배포의 모든 파일을 무효화하려면

  `/*`

  경로의 최대 길이는 4,000자입니다. 경로 내에 와일드카드를 사용할 수 없습니다. 경로 맨 끝에만 사용할 수 있습니다.

  Lambda 함수를 사용하여 URI를 변경하는 경우 파일을 무효화하는 방법에 대한 자세한 내용은 Lambda 함수를 사용하여 URI 변경을 참조하십시오.

  무효화 경로를 제출하는 데 부과되는 요금은 무효화하는 파일 수와 관계없이 단일 객체(`/images/logo.jpg`) 또는 배포와 연결된 모든 파일(`/*`)에 대해 동일합니다. 자세한 내용은 Amazon CloudFront 요금을 참조하십시오.

  무효화 경로가 디렉터리이고 후행 슬래시(/) 포함 여부 등 디렉터리를 지정하는 방식을 표준화하지 않은 경우, 후행 슬래시(/)를 포함한 디렉터리와 포함하지 않은 디렉터리 모두를 무효화하는 것이 좋습니다 (예: `/images` 및 `/images/`).

서명 URL

서명된 URL을 사용하는 경우 물음표(?) 앞의 URL 부분만 포함하여 파일을 무효화합니다.

콘솔을 사용하여 파일 무효화

CloudFront 콘솔을 사용하여 무효화를 생성 및 실행하고 이전에 제출한 무효화의 목록과 개별 무효화에 관한 세부 정보를 표시할 수 있습니다. 또한 기존 무효화를 복사하고 파일 경로의 목록을 편집하며 편집된 무효화를 실행할 수 있습니다. 목록에서 무효화를 제거할 수 없습니다.

- 파일 무효화 (p. 84)
- 기존 무효화 복사, 편집, 재실행 (p. 85)
- 무효화 취소 (p. 85)
- 무효화 목록 (p. 85)
- 모든 무효화에 대한 정보 표시 (p. 86)

파일 무효화

CloudFront 콘솔을 사용하여 파일을 무효화하려면 다음과 같이 합니다.

파일을 무효화하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 파일을 무효화하려는 배포를 선택합니다.
3. Distribution Settings(배포 설정)를 선택합니다.
4. Invalidations(무효화) 탭을 선택합니다.
5. Create Invalidation(무효화 생성)을 선택합니다.
6. 무효화하려는 파일에 대해 한 줄에 하나의 무효화 경로를 입력합니다. 무효화 경로 지정에 대한 자세한 내용은 무효화할 파일 지정 (p. 81) 단원을 참조하십시오.

   Important

   파일 경로를 신중하게 지정합니다. 무효화 요청을 시작한 후에는 취소할 수 없습니다.
7. Invalidate(무효화)를 선택합니다.

기존 무효화 복사, 편집, 재실행

이전에 생성한 무효화를 복사하고 무효화 경로의 목록을 업데이트하여 업데이트된 무효화를 실행할 수 있습니다. 기존 무효화를 복사하고 무효화 경로를 업데이트하여 업데이트된 무효화를 실행하지 않고 저장할 수는 없습니다.

> **Important**
>
> 아직 진행 중인 무효화를 복사하고 무효화 경로의 목록을 업데이트하여 업데이트된 무효화를 실행하는 경우, CloudFront에서는 복사한 무효화를 중단하거나 삭제하지 않습니다. 무효화 경로가 원본 및 복사본에 표시되는 경우 CloudFront에서는 파일에 대해 두 번 무효화를 시도하며, 두 무효화 모두 월별 무료 무효화 최대 횟수로 산정됩니다. 이미 무료 무효화 최대 횟수에 도달한 경우 각 파일의 무효화 두 건에 대해 요금이 청구됩니다. 자세한 내용은 동시 무효화 요청 최대값 (p. 86) 단원을 참조하십시오.

기존 무효화를 복사, 편집, 재실행하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 복사하려는 무효화가 포함된 배포를 선택합니다.
3. Distribution Settings(배포 설정)를 선택합니다.
4. Invalidations(무효화) 탭을 선택합니다.
5. 복사하려는 무효화를 선택합니다.

   복사하려는 무효화가 확실치 않은 경우, 무효화를 하나 선택하고 세부 정보를 선택하여 해당 무효화에 대한 세부 정보를 표시할 수 있습니다.
6. [Copy]를 선택합니다.
7. 해당되는 경우 무효화 경로의 목록을 업데이트합니다.
8. Invalidate(무효화)를 선택합니다.

무효화 취소

CloudFront에 무효화 요청을 제출하면 CloudFront는 몇 초 이내에 모든 엣지 로케이션으로 요청을 전달하며, 각 엣지 로케이션은 즉시 무효화를 처리하기 시작합니다. 따라서 무효화를 제출한 후에는 취소할 수 없습니다.

무효화 목록

CloudFront 콘솔을 사용하여 배포에 대해 생성하고 실행한 최근 100건의 무효화 목록을 표시할 수 있습니다. 100건이 넘는 무효화 목록을 가져오려는 경우 Invalidation List API 작업을 사용합니다. 자세한 내용은 Amazon CloudFront API Reference의 무효화 목록을 참조하십시오.

무효화 목록을 작성하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 무효화 목록을 표시하려는 배포를 선택합니다.
3. Distribution Settings(배포 설정)를 선택합니다.
4. Invalidations(무효화) 탭을 선택합니다.

> **Note**
>
> 목록에서 무효화를 제거할 수 없습니다.

모든 무효화에 대한 정보 표시

배포 ID, 무효화 ID, 무효화의 상태, 무효화가 생성된 날짜 및 시간, 무효화 경로의 전체 목록을 포함한 무효화에 관한 세부 정보를 표시할 수 있습니다.

무효화에 대한 정보를 표시하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 세부 정보를 표시하려는 무효화가 포함된 배포를 선택합니다.
3. Distribution Settings(배포 설정)를 선택합니다.
4. Invalidations(무효화) 탭을 선택합니다.
5. 해당 무효화를 선택합니다.
6. 세부 정보를 선택합니다.

CloudFront API를 사용하여 파일 무효화

CloudFront API를 사용하여 객체를 무효화하고 무효화 관련 정보를 표시하는 방법에 대한 자세한 내용은 Amazon CloudFront API Reference에서 다음 항목을 참조하십시오.

- 파일 무효화: 무효화
- 무효화 목록 가져오기: 무효화 목록
- 특정 무효화 관련 정보 가져오기: 무효화

파일 무효화를 위한 타사 도구

CloudFront에서 제공하는 무효화 방법에 추가로, 여러 타사 도구에서 파일 무효화 방법을 제공하고 있습니다. 도구의 목록은 객체 무효화 (p. 454) 단원을 참조하십시오.

동시 무효화 요청 최대값

파일을 개별적으로 무효화하는 경우, 배포당 한 번에 최대 3,000개의 파일에 대해 진행 중인 무효화 요청을 보유할 수 있습니다. 이것은 최대 3,000개의 파일에 대해 한 번의 무효화 요청을 하거나, 각 파일에 대해 최대 3,000건의 요청을 하거나, 3,000개의 파일을 초과하지 않는 기타 모든 조합이 가능함을 의미합니다. 예를 들어, 각각 100개의 파일을 무효화하는 30건의 무효화 요청을 제출할 수 있습니다. 30건의 무효화 요청 모두가 아직 진행 중인 경우 더 이상의 무효화 요청은 제출할 수 없습니다. 이 최대값을 초과하면 CloudFront에서는 오류 메시지를 반환합니다.

* 와일드카드를 사용하는 경우 한 번에 최대 15개의 무효화 경로에 대해 진행 중인 요청을 보유할 수 있습니다. 또한 배포당 최대 3,000개의 개별 파일에 대해 진행 중인 무효화 요청을 동시에 보유할 수 있습니다. 허용되는 와일드카드 무효화 요청에 대한 최대값은 개별적으로 파일을 무효화하는 데 적용되는 최대값과는 독립적입니다.

파일 무효화에 대한 요금 결제

월별로 제출한 최초 1,000개의 무효화 경로에 대해서는 무료입니다. 월별로 1,000개를 초과하는 각 무효화 경로에 대해서 요금을 지불합니다. 무효화 경로는 단일 파일(예:`/images/logo.jpg`) 또는 여러 파일(예: `/images/*`)에 적용될 수 있습니다. CloudFront가 수천 개의 파일을 무효화하더라도 `*` 와일드카드 수가 하나의 경로로 포함되는 경로입니다.

월별 1,000개의 무효화 경로 최대값은 하나의 AWS 계정으로 생성한 전체 배포에서 무효화 경로의 총 수에 적용됩니다. 예를 들어, AWS 계정 `john@example.com`을 사용하여 세 개의 배포를 만든 경우 지정된 월에

각 배포별로 600개의 무효화 경로를 제출했다면(총 1,800개의 무효화 경로에 해당), AWS에서는 해당 월에 800개의 무효화 경로에 대한 요금을 부과합니다. 무효화 요금에 대한 자세한 내용은 Amazon CloudFront 요금을 참조하십시오. 무효화 경로에 대한 자세한 내용은 무효화 경로 단원을 참조하십시오.

압축된 파일 제공

최종 사용자 요청이 요청 헤더에 `Accept-Encoding: gzip`을 포함하는 경우 특정 유형의 파일을 자동으로 압축하고 압축된 파일을 제공하도록 CloudFront를 구성할 수 있습니다. 콘텐츠가 압축되면 파일 크기가 더 작아지므로— 다운로드 속도가 빨라집니다. 원래 크기의 1/4 이하로 줄어드는 경우도 있습니다. JavaScript 및 CSS 파일의 경우 특히 다운로드 속도가 빨라지면 사용자에게 표시되는 웹 페이지의 렌더링 속도가 빨라집니다. 또한 CloudFront 데이터 전송 비용은 총 데이터 제공량에 따라 달라지므로, 압축된 파일을 제공함으로써 압축되지 않은 파일을 제공할 때에 비해 비용이 절감됩니다.

> **Important**
>
> 최종 사용자 요청에는 요청 헤더에 `Accept-Encoding: gzip`이 포함되어 있어야 하며 그렇지 않은 경우 CloudFront는 요청한 파일을 압축하지 않습니다.

사용자 지정 오리진 또는 Amazon S3 오리진을 사용하는 경우, CloudFront 압축과 관계없이 파일을 압축하도록 오리진을 구성할 수 있습니다. 오리진은 CloudFront가 압축하지 않는 파일 유형을 압축할 수 있습니다. (CloudFront가 압축하는 파일 유형 (p. 89) 단원을 참조하십시오.) 오리진이 CloudFront에 압축된 파일을 반환하는 경우, CloudFront는 `Content-Encoding` 헤더의 값을 기반으로 파일이 압축되었음을 감지하고 파일을 다시 압축하지 않습니다.

주제

- 콘텐츠를 압축하도록 CloudFront 배포 구성 (p. 87)
- CloudFront를 사용하여 콘텐츠 압축 (p. 87)
- 사용자 지정 오리진을 사용하여 콘텐츠 압축 (p. 90)
- Amazon S3 오리진을 사용하여 콘텐츠 압축 (p. 90)
- 오리진 서버가 IIS를 실행 중일 때 압축된 파일 제공 (p. 91)
- 오리진 서버가 NGINX를 실행 중일 때 압축된 파일 제공 (p. 91)

콘텐츠를 압축하도록 CloudFront 배포 구성

배포가 콘텐츠를 압축하도록 구성하려면 다음 방법 중 하나를 사용하여 압축 콘텐츠에 적용하려는 캐시 동작을 업데이트합니다.

- CloudFront 콘솔 – 자동으로 객체를 압축 설정을 업데이트합니다. 자세한 내용은 배포 만들기 (p. 35) 단원을 참조하십시오.
- CloudFront API – `Compress` 요소의 값을 `true`로 변경합니다. 자세한 내용은 CreateDistribution 또는 UpdateDistribution을 참조하십시오.
- AWS SDK 중 하나 – AWS 설명서 페이지에서 SDK 설명서를 참조하십시오.
- AWS CLI – 자세한 내용은 AWS CLI Command Reference의 create-distribution 또는 update-distribution을 참조하십시오.

CloudFront를 사용하여 콘텐츠 압축

CloudFront는 Amazon S3 오리진과 사용자 지정 오리진 모두에 대해 파일을 압축할 수 있습니다. 콘텐츠를 압축하도록 CloudFront를 구성하는 경우, 캐시 동작에서 설정을 지정합니다.

콘텐츠를 압축하도록 CloudFront를 구성하는 경우, CloudFront가 콘텐츠를 제공하는 방법은 다음과 같습니다.

1. CloudFront 배포를 생성 또는 업데이트하고 콘텐츠를 압축하도록 CloudFront를 구성합니다.
2. 최종 사용자가 파일을 요청합니다. 최종 사용자가 `Accept-Encoding: gzip` 헤더를 요청에 추가합니다. 이는 최종 사용자가 압축된 콘텐츠를 지원함을 의미합니다.
3. 이 엣지 로케이션에서 CloudFront는 요청에서 참조된 파일의 압축된 버전의 캐시를 확인합니다.
4. 압축된 파일이 이미 캐시에 있는 경우, CloudFront는 최종 사용자에 파일을 반환하고 나머지 단계를 건너뜁니다.
5. 압축된 파일이 캐시에 없는 경우, CloudFront는 오리진 서버에 요청을 전달하는데, 이것은 Amazon S3 버킷 또는 사용자 지정 오리진 중 하나입니다.

   Note

   CloudFront가 캐시에 압축되지 않은 버전의 파일이 있는 경우에도 오리진에 요청을 전달합니다.
6. 오리진 서버는 CloudFront에 요청한 파일의 압축되지 않은 버전을 반환합니다.
7. CloudFront는 파일을 압축할 수 있는지 확인합니다.
   - 파일은 CloudFront가 압축하는 유형이어야 합니다.
   - 파일 크기는 1,000 ~ 10,000,000바이트 사이여야 합니다.
   - 응답에는 `Content-Length` 헤더가 포함되어 있어 CloudFront에서 파일의 크기가 CloudFront가 압축하는 범위에 있는지 확인할 수 있도록 해야 합니다. `Content-Length` 헤더가 빠진 경우, CloudFront는 파일을 압축하지 않습니다.
   - 응답에 `Content-Encoding` 헤더가 포함되어서는 안 됩니다.
8. 파일을 압축할 수 있는 경우, CloudFront는 압축하고 최종 사용자에 압축된 파일을 반환하고 캐시에 추가합니다.
9. 최종 사용자는 파일을 압축 해제합니다.

다음을 참조하십시오.

CloudFront가 압축하는 파일 유형

CloudFront는 파일을 여러 가지 파일 유형으로 압축합니다. 전체 목록은 CloudFront가 압축하는 파일 유형 (p. 89) 단원을 참조하십시오.

CloudFront가 압축하는 파일 크기

CloudFront는 1,000바이트 ~10,000,000바이트 크기의 파일을 압축합니다.

Content-Length 헤더

오리진에는 응답의 `Content-Length` 헤더가 포함되어 있어 CloudFront에서 파일의 크기가 CloudFront가 압축하는 범위에 있는지 확인할 수 있도록 해야 합니다. `Content-Length` 헤더가 빠진 경우, CloudFront는 파일을 압축하지 않습니다.

Etag 헤더

콘텐츠를 압축하도록 CloudFront를 구성하는 경우, CloudFront는 압축하는 파일에서 `ETag` 응답 헤더를 제거합니다. `ETag` 헤더가 있는 경우, CloudFront와 오리진은 그것을 사용하여 CloudFront 엣지 캐시의 파일 버전이 오리진 서버의 버전과 동일한지 확인할 수 있습니다. 하지만 압축 후 두 버전은 더 이상 동일하지 않게 됩니다. 따라서 압축된 파일이 만료되고 CloudFront가 오리진에 다른 요청을 전달하는 경우, 오리진은 항상 HTTP 상태 코드 304(수정되지 않음) 대신에 CloudFront에 파일을 반환합니다.

파일을 압축하도록 CloudFront를 구성하는 경우 이미 엣지 로케이션에 포함된 콘텐츠

CloudFront는 오리진에서 파일을 가져올 때 각 엣지 로케이션에 파일을 압축합니다. 콘텐츠를 압축하도록 CloudFront를 구성하는 경우, 이미 엣지 로케이션에 포함된 파일은 압축하지 않습니다. 또한 엣지 로케이션에서 파일이 만료되고 CloudFront가 오리진에 파일에 대한 다른 요청을 전달하는 경우, CloudFront는 오리진이 HTTP 상태 코드 304를 반환하는 경우 파일을 압축하지 않기 때문에 엣지 로케이션에는 이미 최신 버전의 파일이 포함되어 있습니다. CloudFront으로 이미 엣지 로케이션에 포함된 파일을 압축하려는 경우, 그 파일을 무효화해야 합니다. 자세한 내용은 파일 무효화 (p. 80) 단원을 참조하십시오.

사용자 지정 오리진이 이미 파일을 압축하도록 구성된 경우

파일을 압축하도록 CloudFront를 구성하고 CloudFront가 역시 파일을 압축하도록 구성된 사용자 지정 오리진에 요청을 전달하는 경우, 사용자 지정 오리진에는 `Content-Encoding` 헤더가 포함되는데, 이를 통해 오리진이 CloudFront에 반환한 파일이 이미 압축되었음을 표시합니다. CloudFront는 최종 사용자에 캐싱된 파일을 반환하고 엣지 로케이션에서 캐싱합니다.

> **Note**
>
> 응답에 `Content-Encoding` 헤더가 포함되어 있는 경우, CloudFront는 값과 관계없이 파일을 압축하지 않습니다.

요청에 Accept-Encoding: gzip이 포함되지 않는 경우

요청에 `Accept-Encoding` 헤더가 빠져 있는 경우, CloudFront는 압축되지 않은 콘텐츠를 제공합니다. `Accept-Encoding` 헤더에 `deflate` 또는 `sdch`와 같은 추가 값이 포함된 경우, CloudFront 에서는 오리진 서버에 요청을 전달하기 전에 이를 제거합니다.

HTTP 1.0을 사용하는 요청

CloudFront에 대한 요청이 HTTP 1.0을 사용하는 경우 CloudFront는 `Accept-Encoding` 헤더를 제거하고 압축되지 않은 콘텐츠를 제공합니다.

CloudFront가 사용 중인 경우

드물긴 해도 CloudFront 엣지 로케이션이 현저한 수준으로 사용 중인 경우, 일부 파일은 압축되지 않을 수 있습니다.

CloudFront가 압축하는 파일 유형

콘텐츠를 압축하도록 CloudFront를 구성하는 경우, CloudFront는 `Content-Type` 헤더에 다음 값이 포함된 파일을 압축합니다.

- `application/dash+xml`
- `application/eot`
- `application/font`
- `application/font-sfnt`
- `application/javascript`
- `application/json`
- `application/opentype`
- `application/otf`
- `application/pkcs7-mime`
- `application/rss+xml`
- `application/truetype`
- `application/ttf`
- `application/vnd.apple.mpegurl`
- `application/vnd.ms-fontobject`
- `application/xhtml+xml`
- `application/xml`
- `application/x-font-opentype`
- `application/x-font-truetype`
- `application/x-font-ttf`
- `application/x-httpd-cgi`
- `application/x-javascript`
- `application/x-mpegurl`

- `application/x-opentype`
- `application/x-otf`
- `application/x-perl`
- `application/x-ttf`
- `font/eot`
- `font/opentype`
- `font/otf`
- `font/ttf`
- `image/svg+xml`
- `text/css`
- `text/csv`
- `text/html`
- `text/javascript`
- `text/js`
- `text/plain`
- `text/richtext`
- `text/tab-separated-values`
- `text/xml`
- `text/x-component`
- `text/x-java-source`
- `text/x-script`
- `vnd.apple.mpegurl`

사용자 지정 오리진을 사용하여 콘텐츠 압축

CloudFront는 gzip을 사용하여 몇 가지 유형의 파일을 압축할 수 있습니다(CloudFront가 압축하는 파일 유형 (p. 89) 참조). 그러나 다른 유형의 파일을 압축하거나 gzip이 아닌 압축 알고리즘(예: brotli)을 사용하려면 자체 서버에서 파일을 압축한 후 해당 파일을 CloudFront를 사용하여 제공하면 됩니다.

CloudFront를 사용하여 gzip 이외의 압축 알고리즘으로 압축된 파일을 제공하려면 `Accept-Encoding` 헤더를 기반으로 캐시하도록 CloudFront를 설정합니다. 이렇게 하면 CloudFront가 `Accept-Encoding` 헤더를 변경하지 않으며 오리진이 최종 사용자에게 해당 압축 파일을 반환할 수 있습니다.

오리진이 CloudFront에 압축된 파일을 반환하는 경우, `Content-Encoding` 헤더를 포함하여 파일이 이미 압축되었음을 CloudFront에 알립니다. 그다음 CloudFront는 압축된 파일을 최종 사용자에게 반환합니다.

Amazon S3 오리진을 사용하여 콘텐츠 압축

Amazon S3를 사용하여 콘텐츠를 저장할 때 gzip 압축 알고리즘을 사용하려면 CloudFront를 사용하여 콘텐츠를 압축할 수 있습니다. 그러나 gzip 대신 또는 gzip 이외에 brotli 등과 같은 다른 압축 알고리즘을 사용하려고 할 수 있습니다.

파일을 압축한 후, 예를 들어 다음과 같이 CloudFront로 파일을 제공할 수 있습니다.

- `Accept-Encoding` 헤더를 기반으로 하여 캐시하도록 CloudFront를 구성합니다.
- 오리진 요청 시 트리거되는 Lambda@Edge 함수를 사용하여 URI를 변경해서 `Accept-Encoding` 헤더에 기초하여 반환하려는 압축 파일을 가리키게 합니다.
- 최종 사용자가 압축된 파일의 형식을 확인할 수 있도록 S3에 저장하는 파일에 콘텐츠 인코딩 메타데이터를 추가합니다. 자세한 내용은 Amazon Simple Storage Service 콘솔 사용 설명서의 S3 객체에 메타데이터를 추가하려면 어떻게 해야 합니까?를 참조하십시오.

오리진 서버가 IIS를 실행 중일 때 압축된 파일 제공

기본적으로 IIS에서는 CloudFront와 같은 프록시 서버를 통해 들어오는 요청에 대해 압축된 콘텐츠를 제공하지 않습니다. IIS를 사용 중인 경우 `httpCompression` 요소를 사용하여 콘텐츠를 압축하도록 IIS를 구성했다면, IIS에서 압축된 콘텐츠를 CloudFront에 반환하도록 `noCompressionForProxies` 속성의 값을 `false`로 변경합니다.

또한 몇 초 단위보다 요청된 빈도가 낮은 객체를 압축한 경우 `frequentHitThreshold` 및 `frequentHitTimePeriod`의 값을 변경해야 할 수 있습니다.

자세한 내용은 Microsoft 웹 사이트에서 IIS 설명서를 참조하십시오.

오리진 서버가 NGINX를 실행 중일 때 압축된 파일 제공

CloudFront에서 오리진 서버에 요청을 전달할 때는 `Via` 헤더를 포함합니다. 이렇게 하면 NGINX에서 요청이 프록시 설정된 것으로 해석되며, 기본적으로 NGINX에서는 프록시 설정된 요청의 압축을 비활성화합니다. NGINX의 버전에 `gzip_proxied` 설정이 포함된 경우, NGINX에서 압축된 콘텐츠를 CloudFront에 반환하도록 이 값을 `any`로 변경합니다. 자세한 내용은 `ngx_http_gzip_module` 모듈에 대한 NGINX 설명서를 참조하십시오.

보안 액세스 구성 및 콘텐츠에 대한 액세스 제한

웹 배포를 위해 CloudFront는 제공하는 콘텐츠의 보안을 유지하기 위한 몇 가지 옵션을 제공합니다. 예를 들어 HTTPS 연결을 구성하거나, AWS WAF를 사용하여 콘텐츠에 대한 액세스를 제어하거나, 특정 콘텐츠 필드에 대해 필드 수준 암호화를 설정할 수 있습니다. 또한 특정 지리적 위치의 사용자가 웹 배포를 통해 배포된 콘텐츠에 액세스하지 못하게 할 수 있습니다. 콘텐츠에 액세스하는 사용자에게 CloudFront 서명된 URL이나 서명된 쿠키를 사용하게 요구하여 프라이빗 콘텐츠에 대한 액세스를 제한할 수도 있습니다.

주제

- CloudFront에서 HTTPS 사용 (p. 92)
- 대체 도메인 이름과 HTTPS 사용 (p. 102)
- 서명된 URL과 서명된 쿠키를 사용하여 프라이빗 콘텐츠 제공 (p. 115)
- 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176)
- AWS AWS WAF를 사용하여 콘텐츠에 대한 액세스 제어 (p. 182)
- 콘텐츠의 지리적 배포 제한 (p. 183)
- 필드 레벨 암호화 사용으로 민감한 데이터 보호 (p. 185)

CloudFront에서 HTTPS 사용

웹 배포에 대해, 최종 사용자가 객체를 요청할 때 HTTPS를 사용할 것을 요구하도록 CloudFront를 구성할 수 있습니다. 이렇게 하면 CloudFront가 최종 사용자와 통신할 때 연결이 암호화됩니다. 또한 CloudFront가 오리진의 객체를 받을 때 HTTPS를 사용하도록 구성할 수 있습니다. 이렇게 하면 CloudFront가 오리진과 통신할 때 연결이 암호화됩니다.

CloudFront가 최종 사용자와 통신할 때 및 오리진과 통신할 때 모두 HTTPS를 요구하도록 구성할 경우, CloudFront에서 객체에 대한 요청을 받을 때 다음과 같은 프로세스가 수행됩니다.

1. 최종 사용자가 HTTPS 요청을 CloudFront에 제출합니다. 최종 사용자와 CloudFront 간에 SSL/TLS 협상이 수행됩니다. 결국에는 최종 사용자가 암호화된 형식을 요청을 제출합니다.
2. 객체가 CloudFront 엣지 캐시에 있는 경우 CloudFront에서 응답을 암호화하여 뷰어에 반환하면 뷰어는 해당 객체를 해독합니다.
3. 객체가 CloudFront 캐시에 있지 않은 경우, CloudFront는 오리진과 SSL/TLS 협상을 수행하며 협상이 완료되면 요청을 암호화된 형식으로 오리진에 전달합니다.
4. 오리진은 요청을 해독한 후 요청한 객체를 암호화하여 CloudFront로 반환합니다.
5. CloudFront는 응답을 해독한 후 다시 암호화하여 최종 사용자에 이 객체를 전달합니다. 또한 CloudFront는 이 객체를 다음 번에 요청될 때 사용할 수 있도록 엣지 캐시에 저장합니다.
6. 최종 사용자가 응답을 해독합니다.

이 프로세스는 오리진이 Amazon S3 버킷, MediaStore 또는 HTTP/S 서버와 같은 사용자 지정 오리진 중 어느것이든 상관없이 기본적으로 동일하게 작동합니다.

> **Note**
>
> SSL 재협상 유형 공격을 차단하기 위해 CloudFront는 최종 사용자와 오리진의 요청에 대한 재협상을 지원하지 않습니다.

최종 사용자와 CloudFront 간에 그리고 CloudFront와 오리진 간에 HTTPS를 요청하는 방법을 알아보려면 다음 항목을 참조하십시오.

주제

- 최종 사용자와 CloudFront 간의 통신에 HTTPS 요구 (p. 93)
- CloudFront와 사용자 지정 오리진 간의 통신에 HTTPS 요구 (p. 94)
- CloudFront와 Amazon S3 오리진 간의 통신에 HTTPS 필요 (p. 97)
- 지원되는 프로토콜과 암호 (p. 98)
- HTTPS 연결 요금 (p. 102)

최종 사용자와 CloudFront 간의 통신에 HTTPS 요구

CloudFront 배포에 하나 이상의 캐시 동작을 구성하여 최종 사용자와 CloudFront 간의 통신에 HTTPS를 요구할 수 있습니다. 또한, CloudFront가 일부 객체에 대해서만 HTTPS를 요구하도록 하기 위해 HTTP 및 HTTPS 모두를 허용하는 캐시 동작을 한 개 이상 구성할 수 있습니다. 구성 단계는 객체 URL에 사용하는 도메인 이름에 따라 다릅니다.

- CloudFront가 배포에 할당한 도메인 이름(예: d111111abcdef8.cloudfront.net)을 사용할 경우 한 개 이상의 캐시 동작에 대해 Viewer Protocol Policy(최종 사용자 프로토콜 정책) 설정을 변경하여 HTTPS 통신을 요구하도록 합니다. 이 구성에서 CloudFront는 SSL/TLS 인증서를 제공합니다.

  CloudFront 콘솔을 사용하여 Viewer Protocol Policy(최종 사용자 프로토콜 정책) 값을 변경하려면 이 단원의 절차를 참조하십시오.

  CloudFront API를 사용하여 `ViewerProtocolPolicy` 요소의 값을 변경하는 방법을 보려면 Amazon CloudFront API Reference의 UpdateDistribution 단원을 참조하십시오.
- example.com 등의 자체 도메인 이름을 사용할 경우 몇 가지 CloudFront 설정을 변경해야 합니다. 또한 AWS Certificate Manager(ACM)에서 제공한 SSL/TLS 인증서를 사용하거나, 다른 인증 기관의 인증서를 ACM 또는 IAM 인증서 스토어로 가져와야 합니다. 자세한 내용은 대체 도메인 이름과 HTTPS 사용 (p. 102) 단원을 참조하십시오.

> **Note**
>
> CloudFront가 오리진에서 객체를 받을 때 최종 사용자가 CloudFront로부터 받는 객체가 암호화되도록 하려면 CloudFront와 오리진 간에 항상 HTTPS를 사용하세요. 최근에 CloudFront와 오리진 간의 연결을 HTTP에서 HTTPS로 변경한 경우 CloudFront 엣지 로케이션의 객체를 무효화할 것을 권장합니다. CloudFront는 최종 사용자가 사용한 프로토콜(HTTP 또는 HTTPS)이, CloudFront가 객체를 받는 데 사용한 프로토콜과 일치하는지와 상관없이 최종 사용자에 객체를 반환합니다. 배포에서 객체 제거 또는 대체에 대한 자세한 내용은 CloudFront가 배포하는 콘텐츠 추가, 제거 또는 바꾸기 (p. 77) 단원을 참조하십시오.

하나 이상의 캐시 동작에 대해 최종 사용자와 CloudFront 간에 HTTPS를 요구하려면 다음과 같이 하세요.

CloudFront에서 최종 사용자와 CloudFront 간에 HTTPS를 요구하도록 구성하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. CloudFront 콘솔의 상단 창에서 업데이트할 배포의 ID를 선택합니다.
3. 동작 탭에서 업데이트할 캐시 동작을 선택한 후 편집을 선택합니다.

4. Viewer Protocol Policy(최종 사용자 프로토콜 정책)에 다음 값 중 하나를 지정합니다.

   Redirect HTTP to HTTPS

   최종 사용자가 두 프로토콜을 모두 사용할 수 있습니다. HTTP `GET` 및 `HEAD` 요청은 HTTPS 요청으로 자동 리디렉션됩니다. CloudFront는 HTTP 상태 코드 301(영구 이동됨)을 새로운 HTTPS URL과 함께 반환합니다. 그러면 최종 사용자는 HTTPS URL을 사용하여 이 요청을 CloudFront에 다시 제출합니다.

   Important

   HTTP를 통해 `POST`, `PUT`, `DELETE`, `OPTIONS` 또는 `PATCH`를 HTTP - HTTPS 캐시 동작 및 HTTP 1.1 이상의 요청 프로토콜 버전으로 보내면, CloudFront는 HTTP 상태 코드 307(임시 리디렉션)과 함께 HTTPS 위치로 요청을 리디렉션합니다. 이렇게 하면 요청이 동일한 메서드와 본문 페이로드를 사용하여 새 위치로 다시 전송됩니다.
   `POST`, `PUT`, `DELETE`, `OPTIONS` 또는 `PATCH` 요청을 HTTP - HTTPS 캐시 동작을 통해 HTTP 1.1 미만인 요청 프로토콜 버전으로 보내면, CloudFront는 HTTP 상태 코드 403(사용할 수 없음)을 반환합니다.

   최종 사용자가 HTTPS 요청으로 리디렉션되는 HTTP 요청을 만들 경우 CloudFront에서 두 요청 모두에 대해 요금을 부과합니다. HTTP 요청의 경우에는 CloudFront에서 최종 사용자에게 반환되는 요청 및 헤더에만 요금이 부과됩니다. HTTPS 요청의 경우에는 요청, 헤더, 그리고 오리진에 의해 반환된 객체에 대해 요금이 부과됩니다.

   HTTPS Only

   최종 사용자가 HTTPS를 사용할 경우에만 콘텐츠에 액세스할 수 있습니다. 최종 사용자가 HTTPS 요청 대신에 HTTP 요청을 보내면 CloudFront는 HTTP 상태 코드 403(금지됨)을 반환하고 객체는 반환하지 않습니다.
5. 예, 편집합니다를 선택합니다.
6. 최종 사용자와 CloudFront 간에 HTTPS를 요구하려는 캐시 동작에 대해 3-5단계를 반복합니다.
7. 업데이트한 구성을 프로덕션 환경에서 사용하기 전에 다음 사항을 확인합니다.
   - 각 캐시 동작의 경로 패턴이 최종 사용자에 HTTPS를 사용하도록 지정한 요청에만 적용되는가.
   - 캐시 동작이 CloudFront에서 평가하도록 할 순서대로 나열되었는가. 자세한 내용은 경로 패턴 (p. 44) 단원을 참조하십시오.
   - 캐시 동작이 올바른 오리진에 요청을 라우팅하는가.

CloudFront와 사용자 지정 오리진 간의 통신에 HTTPS 요구

CloudFront와 사용자 지정 오리진 간의 통신을 위해 HTTPS를 요청하려는 경우 수행하는 단계는 CloudFront에서 배포에 할당한 도메인 이름(예: d111111abcdef8.cloudfront.net)을 사용하는지 또는 자신의 대체 도메인 이름(예: example.com)을 사용하는지에 따라 다릅니다.

Note

Amazon S3 버킷을 오리진으로 사용하고 버킷을 웹 사이트 엔드포인트로 구성하는 경우 이 단원의 지침을 따르십시오. 이렇게 구성하지 않은 경우, Amazon S3 버킷을 HTTPS와 함께 사용하려면 CloudFront와 Amazon S3 오리진 간의 통신에 HTTPS 필요 (p. 97) 단원을 참조하십시오.

기본 CloudFront 도메인 사용

CloudFront에서 배포에 할당한 도메인 이름을 객체의 URL에 사용하는 경우(예: https://d111111abcdef8.cloudfront.net/logo.jpg) 이 주제의 절차에 따라 HTTPS를 요청하여 다음을 수행할 수 있습니다.

- 배포의 특정 오리진에 대한 오리진 프로토콜 정책 설정 변경
- 사용자 지정 오리진 서버에 SSL/TLS 인증서를 설치합니다(Amazon S3 오리진을 사용하는 경우 이 단계가 필요 없음).

대체 도메인 이름 사용

기본 도메인 이름을 배포에 사용하는 대신, example.com과 같이 작업하기 쉬운 대체 도메인 이름을 추가할 수 있습니다.

대체 도메인 이름을 사용할 때 통신을 위한 HTTPS를 요청하려면 대체 도메인 이름과 HTTPS 사용 (p. 102)의 단계와 지침을 따르십시오.

주제

- CloudFront 설정 변경 (p. 95)
- 사용자 지정 오리진 서버에 SSL/TLS 인증서 설치 (p. 96)
- RSA 및 ECDSA 암호 소개 (p. 96)

CloudFront 설정 변경

다음 절차는 CloudFront에서 HTTPS를 사용하여 Elastic Load Balancing 로드 밸런서, Amazon EC2 인스턴스 또는 다른 사용자 지정 오리진과 통신하도록 구성하는 방법을 보여 줍니다. CloudFront API를 사용한 웹 배포 업데이트에 대한 자세한 내용은 Amazon CloudFront API Reference의 UpdateDistribution 단원을 참조하십시오.

CloudFront에서 CloudFront와 사용자 지정 오리진 간에 HTTPS를 요구하도록 구성하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. CloudFront 콘솔의 상단 창에서 업데이트할 배포의 ID를 선택합니다.
3. 오리진 탭에서 업데이트할 오리진을 선택하고 편집을 선택합니다.
4. 다음 설정을 업데이트합니다.

   오리진 프로토콜 정책

   배포의 해당 오리진에 대해 오리진 프로토콜 정책을 변경합니다.

   - HTTPS Only(HTTPS만 해당) – CloudFront에서 사용자 지정 오리진과 통신할 때 HTTPS만 사용합니다.
   - Match Viewer – CloudFront에서는 뷰어 요청의 프로토콜에 따라 HTTP 또는 HTTPS를 사용하여 오리진과 통신합니다. 예를 들면 오리진 프로토콜 정책에 대해 Match Viewer를 선택하고 최종 사용자가 HTTPS를 사용하여 CloudFront에서 객체를 요청하는 경우, CloudFront에서도 HTTPS를 사용하여 해당 요청을 오리진으로 전달합니다.

     Viewer Protocol Policy(최종 사용자 프로토콜 정책)에서 Redirect HTTP to HTTPS(HTTP를 HTTPS로 재지정) 또는 HTTPS Only(HTTPS만 해당)를 지정한 경우에만 Match Viewer를 선택합니다.

     최종 사용자가 HTTP 및 HTTPS 프로토콜 모두를 사용하여 요청하더라도 CloudFront에서는 한 번만 객체를 캐싱합니다.

   오리진 SSL 프로토콜

   배포의 해당 오리진에 대해 오리진 SSL 프로토콜을 선택합니다. SSLv3 프로토콜이 덜 안전하므로 사용자가 TLSv1 이상을 지원하지 않는 경우에만 SSLv3을 선택하는 것이 좋습니다. TLSv1 핸드쉐이크는 SSLv3의 이전 버전 및 이후 버전과 모두 호환되지만, TLSv1.1 및 TLSv1.2는 호환되지 않습니다. SSLv3을 선택하면 CloudFront는 SSLv3 핸드셰이크 요청만 보냅니다.

5. 예, 편집합니다를 선택합니다.
6. CloudFront와 사용자 지정 오리진 간에 HTTPS를 요구하려는 다른 오리진에 대해 3-5단계를 반복합니다.
7. 업데이트한 구성을 프로덕션 환경에서 사용하기 전에 다음 사항을 확인합니다.
   - 각 캐시 동작의 경로 패턴이 최종 사용자에 HTTPS를 사용하도록 지정한 요청에만 적용되는가.
   - 캐시 동작이 CloudFront에서 평가하도록 할 순서대로 나열되었는가. 자세한 내용은 경로 패턴 (p. 44) 단원을 참조하십시오.
   - 캐시 동작은 변경한 오리진 프로토콜 정책에 따라 요청을 오리진으로 라우팅합니다.

사용자 지정 오리진 서버에 SSL/TLS 인증서 설치

사용자 지정 오리진에서 다음 소스의 SSL/TLS 인증서를 사용할 수 있습니다.

- 오리진이 Elastic Load Balancing 로드 밸런서일 경우 AWS Certificate Manager(ACM)에서 제공한 인증서를 사용할 수 있습니다. 또한 신뢰할 수 있는 다른 인증 기관에서 서명한 인증서 및 ACM으로 가져온 인증서를 사용할 수도 있습니다.
- ELB 로드 밸런서 외의 오리진에서는 Comodo, DigiCert, Symantec 같이 신뢰할 수 있는 타사 인증 기관(CA)에서 서명한 인증서를 사용해야 합니다.

CloudFront가 HTTPS를 사용하여 오리진과 통신하면 CloudFront는 인증서가 신뢰할 수 있는 인증 기관에 의해 발급되었는지 확인합니다. CloudFront는 Mozilla가 지원하는 것과 동일한 인증 기관을 지원합니다. 최신 목록은 Mozilla에 포함된 CA 인증서 목록을 참조하십시오. CloudFront와 오리진 간의 HTTPS 통신에는 자체 서명한 인증서를 사용할 수 없습니다.

> **Important**
>
> 오리진 서버에서 만료된 인증서, 잘못된 인증서 또는 자체 서명된 인증서를 반환하는 경우 또는 오리진 서버에서 잘못된 순서로 인증서 체인을 반환하는 경우에는 CloudFront에서 TCP 연결을 끊고 HTTP 상태 코드 502(잘못된 게이트웨이)를 반환하며 `X-Cache` 헤더를 `Error from cloudfront`로 설정합니다. 또한 중간 인증서를 포함하여 인증서의 전체 체인이 없는 경우, CloudFront는 TCP 연결을 끊습니다.

오리진에서 반환된 인증서는 사용자의 배포에서 해당 오리진에 대해 오리진 도메인 이름을 지정한 도메인을 포함해야 합니다. 또한 `Host` 헤더를 오리진으로 전달하도록 CloudFront를 구성한 경우 원본은 `Host` 헤더의 도메인과 일치하는 인증서로 응답해야 합니다.

RSA 및 ECDSA 암호 소개

통신 연결의 암호화 강도는 오리진 서버의 인증서로 선택한 알고리즘의 키 크기와 강도에 따라 다릅니다. CloudFront는 사용자 지정 오리진과의 연결을 위해 RSA와 ECDSA(Elliptic Curve Digital Signature Algorithm) 두 개의 옵션을 지원합니다.

CloudFront에서 지원되는 RSA 및 ECDSA 암호의 목록은 CloudFront와 오리진 간 통신에 지원되는 SSL/TLS 프로토콜 및 암호 (p. 101) 단원을 참조하십시오.

RSA 암호의 작동 방식

CloudFront와 오리진 서버는 일반적으로 SSL/TLS 종료를 위해 RSA 2048비트 비대칭 키를 사용합니다. RSA 알고리즘은 큰 두 개의 소수로 이루어진 PRODUCT 함수와 함께 퍼블릭 키를 생성하기 위해 추가된 또 다른 숫자를 사용합니다. 프라이빗 키는 관련 숫자입니다. RSA의 강도는 키의 암호를 해독할 때 큰 두 개의 소수로 이루어진 PRODUCT 함수를 소인수 분해하는 것을 얼마나 난해하게 추정하느냐에 따라 다릅니다. 한편, 컴퓨터 기술의 발전으로 RSA 알고리즘이 약해졌습니다. 왜냐하면 컴퓨터 계산 속도가 빨라질수록 훨씬 손쉽게 암호를 해독할 수 있기 때문입니다.

RSA를 계속 사용하면서 암호화 강도를 유지하고 싶다면 RSA 키의 크기를 늘리는 것이 한 방법이 될 수 있습니다. 그러나 더 큰 키를 사용할수록 암호화를 위한 컴퓨팅 비용이 증가하기 때문에 이 접근 방법을 손쉽게 확장할 수 없습니다.

ECDSA 암호의 작동 방식

또는 ECDSA 인증서를 사용할 수 있습니다. ECDSA는 RSA보다 훨씬 복잡해서 해결이 어려운 수학적 문제에 대해서도 보안을 보장하기 때문에 ECDSA 암호를 해독하는 데 컴퓨터 처리 시간이 더 많이 걸립니다. ECDSA는 기반이 ECDLP라는 것이 알려진 경우에 임의 타원 곡선의 이산 로그를 해독하는 것이 어렵다는 원칙을 기반으로 합니다. 따라서 더 짧은 길이의 키를 사용하여 크기가 훨씬 큰 키에서 RSA를 사용하는 것과 맞먹는 보안을 달성할 수 있습니다.

크기가 더 작은 ECDSA의 키를 사용하면 보안을 강화하는 것 외에도 알고리즘을 신속하게 컴퓨팅하고 더 작은 디지털 인증서를 사용하며 SSL/TLS 핸드쉐이크 동안 전송되는 비트 수를 줄일 수 있습니다. 그 결과, 크기가 더 작은 키는 오리진 서버에서 SSL/TLS 종료를 위해 디지털 인증서를 생성 및 로그인하는 데 걸리는 시간을 줄일 수 있습니다. 따라서 크기가 작은 키를 사용하면 암호화에 필요한 컴퓨팅 주기를 줄이고 서버 리소스들을 다른 작업의 처리에 투입함으로써 처리량을 늘릴 수 있습니다.

RSA 및 ECDSA 암호 중 선택

예를 들어, 2048비트 RSA와 256비트 ECDSA(nistp256)의 비교를 위해 샘플 테스트를 실행한 결과, nistp256 옵션이 2048비트 RSA보다 속도는 95%더 빠르고 3072비트 RSA와 동일한 보안 강도를 제공하는 것으로 나타났습니다.

CloudFront는 SSL/TLS 연결을 위해 RSA를 계속 지원하고 있습니다. 그러나 오리진 서버의 SSL/TLS 인증에서 현재 사용되는 암호화의 강도가 우려되는 경우에는 ECDSA가 더 나은 선택일 수 있습니다. ECDSA는 디지털 인증서를 활성화해야 하는 번거로움이 있는 반면 보안 이점이 크기 때문에 의사 결정 시 이 점을 중요하게 고려해야 합니다. 보다 강한 암호화를 사용할 수 있다는 점 외에도 오리진 서버에서 ECDSA를 사용함으로써 암호화 컴퓨팅 비용을 줄일 수 있다는 것도 부가적인 이점입니다.

ECDSA 암호 사용

CloudFront와 오리진 간의 통신을 위해 ECDSA를 사용하려면 다음과 같이 수행하십시오.

1. 지원되는 곡선(prime256v1 또는 secp384r1) 중 하나를 사용하여 프라이빗 키를 생성합니다.
2. 신뢰할 수 있는 인증 기관을 통해 X.509 PEM 형식으로 ECDSA 디지털 인증서를 생성합니다.
3. ECDSA 인증서를 선호하도록 오리진을 설정합니다.

ECDSA를 사용하면 CloudFront 콘솔 또는 API의 설정을 변경할 필요가 없기 때문에 추가 비용이 들지 않습니다.

CloudFront와 Amazon S3 오리진 간의 통신에 HTTPS 필요

오리진이 Amazon S3 버킷인 경우 CloudFront와 통신을 위해 HTTPS를 사용하는 옵션은 버킷을 사용하는 방식에 따라 달라집니다. 웹 사이트 엔드포인트로 Amazon S3 버킷이 구성되어 있는 경우, Amazon S3에서 해당 구성으로 HTTPS 연결을 지원하지 않으므로 HTTPS를 사용하여 오리진과 통신하도록 CloudFront를 구성할 수 없습니다.

오리진이 HTTPS 통신을 지원하는 Amazon S3 버킷일 경우 CloudFront은(는) 항상 최종 사용자가 요청을 전송하는 데 사용된 프로토콜을 사용하여 S3로 요청을 전송합니다. 오리진 프로토콜 정책 (p. 41) 설정의 기본 설정은 Match Viewer이며 변경할 수 없습니다.

CloudFront와 Amazon S3 간의 통신에 HTTPS가 필요한 경우, Viewer Protocol Policy(최종 사용자 프로토콜 정책) 값을 Redirect HTTP to HTTPS(HTTP를 HTTPS로 재지정) 또는 HTTPS Only(HTTPS만 해당)로 변경해야 합니다. 이 단원의 나머지 절차에서는 CloudFront 콘솔을 사용하여 Viewer Protocol

Policy(최종 사용자 프로토콜 정책) 값을 변경하는 방법을 설명합니다. CloudFront API를 사용하여 웹 배포의 `ViewerProtocolPolicy` 요소를 업데이트하는 방법에 대한 자세한 내용은 Amazon CloudFront API Reference의 UpdateDistribution 단원을 참조하십시오.

HTTPS 통신을 지원하는 Amazon S3 버킷과 함께 HTTPS를 사용하면 Amazon S3이(가) SSL/TLS 인증서를 제공하므로 사용자가 제공할 필요가 없습니다.

CloudFront이(가) Amazon S3 오리진에 대해 HTTPS를 요구하도록 구성하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. CloudFront 콘솔의 상단 창에서 업데이트할 배포의 ID를 선택합니다.
3. 동작 탭에서 업데이트할 캐시 동작을 선택한 후 편집을 선택합니다.
4. Viewer Protocol Policy(최종 사용자 프로토콜 정책)에 다음 값 중 하나를 지정합니다.

   Redirect HTTP to HTTPS

   최종 사용자가 두 프로토콜 모두 사용할 수 있지만 HTTP 요청은 HTTPS 요청으로 자동 리디렉션됩니다. CloudFront는 HTTP 상태 코드 301(영구 이동됨)을 새로운 HTTPS URL과 함께 반환합니다. 그러면 최종 사용자는 HTTPS URL을 사용하여 이 요청을 CloudFront에 다시 제출합니다.

   > **Important**
   >
   > CloudFront는 HTTP에서 HTTPS로 `DELETE`, `OPTIONS`, `PATCH`, `POST` 또는 `PUT` 요청을 리디렉션하지 않습니다. HTTPS에 리디렉션할 캐시 동작을 구성하는 경우, CloudFront는 해당 캐시 동작에 대한 HTTP `DELETE`, `OPTIONS`, `PATCH`, `POST` 또는 `PUT` 요청에 HTTP 상태 코드 403(금지됨)으로 응답합니다.

   최종 사용자가 HTTPS 요청으로 리디렉션되는 HTTP 요청을 만들 경우 CloudFront에서 두 요청 모두에 대해 요금을 부과합니다. HTTP 요청의 경우에는 CloudFront에서 최종 사용자에게 반환되는 요청 및 헤더에만 요금이 부과됩니다. HTTPS 요청의 경우에는 요청, 헤더, 그리고 오리진에 의해 반환된 객체 모두에 요금이 부과됩니다.

   HTTPS Only

   최종 사용자가 HTTPS를 사용할 경우에만 콘텐츠에 액세스할 수 있습니다. 최종 사용자가 HTTPS 요청 대신에 HTTP 요청을 보내면 CloudFront는 HTTP 상태 코드 403(금지됨)을 반환하고 객체는 반환하지 않습니다.
5. 예, 편집합니다를 선택합니다.
6. 최종 사용자와 CloudFront 간에 그리고 CloudFront와 S3 간에 HTTPS를 요구하려는 캐시 동작에 대해 3~5단계를 반복합니다.
7. 업데이트한 구성을 프로덕션 환경에서 사용하기 전에 다음 사항을 확인합니다.
   - 각 캐시 동작의 경로 패턴이 최종 사용자에 HTTPS를 사용하도록 지정한 요청에만 적용되는가.
   - 캐시 동작이 CloudFront에서 평가하도록 할 순서대로 나열되었는가. 자세한 내용은 경로 패턴 (p. 44) 단원을 참조하십시오.
   - 캐시 동작이 올바른 오리진에 요청을 라우팅하는가.

지원되는 프로토콜과 암호

최종 사용자와 CloudFront 간 통신과 CloudFront와 오리진 간 통신을 위해 HTTPS 설정을 선택할 수 있습니다.

- 최종 사용자와 CloudFront 간 – 최종 사용자와 CloudFront 간에 HTTPS가 필요할 경우 최종 사용자와 CloudFront가 통신하는 데 사용할 수 있는 프로토콜을 결정하는 보안 정책도 선택합니다. 또한 보안 정책은 CloudFront가 최종 사용자로 반환하는 콘텐츠를 암호화하는 데 사용할 수 있는 암호를 결정합니다.

- CloudFront와 오리진 간 – CloudFront와 오리진 간에 HTTPS가 필요할 경우 CloudFront와 오리진이 통신하는 데 사용하는 프로토콜도 선택합니다. 선택한 프로토콜에 따라 오리진이 CloudFront로 반환되는 콘텐츠를 암호화하는 데 사용할 수 있는 암호가 결정됩니다.

주제

- 최종 사용자와 CloudFront 간 통신에 지원되는 SSL/TLS 프로토콜 및 암호 (p. 99)
- CloudFront와 오리진 간 통신에 지원되는 SSL/TLS 프로토콜 및 암호 (p. 101)

최종 사용자와 CloudFront 간 통신에 지원되는 SSL/TLS 프로토콜 및 암호

최종 사용자와 CloudFront 간에 HTTPS를 요구할지 여부를 선택하려면 Viewer Protocol Policy (p. 45)에 해당 값을 지정합니다.

HTTPS가 필요할 경우 CloudFront가 HTTPS 연결에 사용할 보안 정책도 선택합니다. 보안 정책은 다음 두 가지 설정을 결정합니다.

- CloudFront가 최종 사용자와 통신하는 데 사용하는 최소 SSL/TLS 프로토콜
- CloudFront가 최종 사용자로 반환하는 콘텐츠를 암호화할 때 사용할 수 있는 암호입니다.

사용자가 TLSv1.2를 지원하지 않는 브라우저 또는 디바이스를 사용하지 않을 경우 TLSv1.2_2018을 지정하는 것이 좋습니다.

보안 정책을 선택하려면 보안 정책 (p. 53)에 해당 값을 지정합니다. 다음 표에는 CloudFront가 각 보안 정책에 사용할 수 있는 프로토콜 및 암호가 나와 있습니다.

최종 사용자는 지원되는 암호 중 하나 이상을 지원하여 CloudFront와의 HTTPS 연결을 설정해야 합니다. AWS Certificate Manager에서 SSL/TLS 인증서를 사용하는 경우, 최종 사용자는 *-RSA-* 암호화 중 하나를 지원해야 합니다. CloudFront에서는 최종 사용자가 지원하는 암호 중에서 나열된 순서대로 암호를 선택합니다. 또한 OpenSSL 및 RFC 암호 이름 (p. 100) 단원도 참조하십시오.

	보안 정책				
지원되는 SSL/TLS 프로토콜					
TLSv1.2	◆	◆	◆	◆	◆
TLSv1.1	◆	◆	◆	◆	
TLSv1	◆	◆	◆		
SSLv3	◆				
지원되는 암호					
ECDHE-RSA-AES128-GCM-SHA256	◆	◆	◆	◆	◆
ECDHE-RSA-AES128-SHA256	◆	◆	◆	◆	◆
ECDHE-RSA-AES128-SHA	◆	◆	◆	◆	
ECDHE-RSA-AES256-GCM-SHA384	◆	◆	◆	◆	◆
ECDHE-RSA-AES256-SHA384	◆	◆	◆	◆	◆
ECDHE-RSA-AES256-SHA	◆	◆	◆	◆	

	보안 정책				
AES128-GCM-SHA256	◆	◆	◆	◆	◆
AES256-GCM-SHA384	◆	◆	◆	◆	◆
AES128-SHA256	◆	◆	◆	◆	◆
AES256-SHA	◆	◆	◆	◆	
AES128-SHA	◆	◆	◆	◆	
DES-CBC3-SHA	◆	◆			
RC4-MD5	◆				

OpenSSL 및 RFC 암호 이름

OpenSSL 및 IETF RFC 5246인 TLS(전송 계층 보안) 프로토콜 버전 1.2는 동일한 암호에 대해 서로 다른 이름을 사용합니다. 다음 표에는 각 암호의 RFC 이름에 OpenSSL 이름이 매핑되어 있습니다.

OpenSSL 암호화 이름	RFC 암호화 이름
ECDHE-RSA-AES128-GCM-SHA256	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
ECDHE-RSA-AES128-SHA256	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
ECDHE-RSA-AES128-SHA	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
ECDHE-RSA-AES256-GCM-SHA384	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
ECDHE-RSA-AES256-SHA384	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
ECDHE-RSA-AES256-SHA	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
AES128-GCM-SHA256	TLS_RSA_WITH_AES_128_GCM_SHA256
AES256-GCM-SHA384	TLS_RSA_WITH_AES_256_GCM_SHA384
AES128-SHA256	TLS_RSA_WITH_AES_128_CBC_SHA256
AES256-SHA	TLS_RSA_WITH_AES_256_CBC_SHA
AES128-SHA	TLS_RSA_WITH_AES_128_CBC_SHA
DES-CBC3-SHA	TLS_RSA_WITH_3DES_EDE_CBC_SHA
RC4-MD5	TLS_RSA_WITH_RC4_128_MD5

CloudFront와 오리진 간 통신에 지원되는 SSL/TLS 프로토콜 및 암호

CloudFront와 오리진 간에 HTTPS 필요를 선택하면 어떤 SSL/TLS 프로토콜이 보안 연결을 허용하는지 확인하고 CloudFront에서 지원되는 어떤 암호(아래 표 참조)든 선택하여 오리진에 대해 HTTPS 연결을 설정할 수 있습니다.

CloudFront는 이 단원에 나와 있는 ECDSA 또는 RSA 암호를 사용하여 오리진 서버로 HTTPS 요청을 전달할 수 있습니다. 오리진 서버가 오리진에 대해 HTTPS 연결을 설정하려면 CloudFront에 대해 이들 암호화 중 하나 이상을 지원해야 합니다. CloudFront가 지원하는 두 가지 유형의 암호에 대한 자세한 내용은 RSA 및 ECDSA 암호 소개 (p. 96) 단원을 참조하십시오.

> **Note**
>
> 다음 곡선은 타원 곡선 기반 암호에서 지원됩니다.
>
> - prime256v1
> - secp384r1

OpenSSL 및 IETF RFC 5246인 TLS(전송 계층 보안) 프로토콜 버전 1.2는 동일한 암호에 대해 서로 다른 이름을 사용합니다. 다음 표에는 각 암호의 RFC 이름에 OpenSSL 이름이 매핑되어 있습니다.

지원되는 RSA 암호

CloudFront는 오리진과의 연결을 위해 다음과 같은 RSA 암호를 지원합니다.

OpenSSL 암호화 이름	RFC 암호화 이름
ECDHE-RSA-AES128-SHA256	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
ECDHE-RSA-AES256-SHA384	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
AES256-SHA	TLS_RSA_WITH_AES_256_CBC_SHA
AES128-SHA	TLS_RSA_WITH_AES_128_CBC_SHA
DES-CBC3-SHA	TLS_RSA_WITH_3DES_EDE_CBC_SHA
RC4-MD5	TLS_RSA_WITH_RC4_128_MD5

지원되는 ECDSA 암호

CloudFront는 오리진과의 연결을 위해 다음과 같은 ECDSA 암호를 지원합니다.

OpenSSL 암호화 이름	RFC 암호화 이름
ECDHE-ECDSA-AES256-GCM-SHA384	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
ECDHE-ECDSA-AES256-SHA384	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384

OpenSSL 암호화 이름	RFC 암호화 이름
ECDHE-ECDSA-AES256-SHA	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
ECDHE-ECDSA-AES128-GCM-SHA256	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
ECDHE-ECDSA-AES128-SHA256	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
ECDHE-ECDSA-AES128-SHA	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA

HTTPS 연결 요금

HTTPS 요청에는 항상 추가 요금이 발생할 수 있습니다. 자세한 내용은 Amazon CloudFront 요금을 참조하십시오.

대체 도메인 이름과 HTTPS 사용

파일의 URL에 자체 도메인 이름(예 `https://www.example.com/image.jpg`)을 사용하고 최종 사용자가 HTTPS를 사용하게 하려면 이 단원에서 설명하는 단계를 추가로 수행해야 합니다. (기본 CloudFront 배포 도메인 이름을 URL에 사용하는 경우에는(예: `https://d111111abcdef8.cloudfront.net/image.jpg`) 최종 사용자와 CloudFront 간의 통신에 HTTPS 요구 (p. 93) 주제의 지침을 따르십시오.)

> **Important**
>
> 인증서를 배포에 추가하는 경우 CloudFront는 인증서를 모든 엣지 로케이션으로 즉시 전파합니다. 새 엣지 로케이션를 사용할 수 있게 되면 CloudFront에서는 인증서를 그러한 새 위치로도 전파합니다. CloudFront에서 인증서를 전파하는 엣지 로케이션을 제한할 수는 없습니다.

주제

- CloudFront에서 HTTPS 요청을 제공하는 방식 선택 (p. 102)
- CloudFront에서 SSL/TLS 인증서를 사용하기 위한 요구 사항 (p. 104)
- CloudFront에서 SSL/TLS 인증서 사용 시의 할당량(최종 사용자와 CloudFront 간의 HTTPS만 해당) (p. 107)
- 대체 도메인 이름과 HTTPS 구성 (p. 108)
- SSL/TLS 인증서에서 퍼블릭 키 크기 확인 (p. 111)
- SSL/TLS 인증서에 대한 할당량 증가 (p. 112)
- SSL/TLS 인증서 회전 (p. 113)
- 사용자 지정 SSL/TLS 인증서에서 기본 CloudFront 인증서로 되돌리기 (p. 114)
- 사용자 지정 SSL/TLS 인증서를 전용 IP 주소 사용에서 SNI 사용으로 전환 (p. 114)

CloudFront에서 HTTPS 요청을 제공하는 방식 선택

최종 사용자가 HTTPS를 사용하고 파일에 다른 도메인 이름을 사용하게 하려면 CloudFront에서 HTTPS 요청을 제공하는 방식에 대해 다음 옵션 중 하나를 선택해야 합니다.

- 서버 이름 표시(SNI) 사용 - 권장
- 각 엣지 로케이션에서 전용 IP 주소 사용

이 단원에서는 각 옵션이 어떻게 작동하는지를 설명합니다.

SNI를 사용하여 HTTPS 요청 제공(대부분 클라이언트에 적용)

서버 이름 표시(SNI)는 2010년 이후 출시된 브라우저와 클라이언트에서 지원되는 TLS 프로토콜의 확장 기능입니다. SNI를 사용하여 HTTPS 요청을 제공하도록 CloudFront를 구성한 경우 CloudFront에서는 대체 도메인 이름을 각 엣지 로케이션의 IP 주소와 연결합니다. 최종 사용자가 HTTPS 콘텐츠 요청을 제출하면 DNS는 이 요청을 올바른 엣지 로케이션의 IP 주소로 라우팅합니다. 도메인 이름의 IP 주소는 SSL/TLS 핸드셰이크 협상 중에 결정됩니다(IP 주소는 배포 전용이 아님).

HTTPS 연결 설정 과정 초기에 SSL/TLS 협상이 발생합니다. CloudFront에서 요청과 관련된 도메인을 즉시 확인할 수 없으면 연결이 끊어집니다. SNI를 지원하는 최종 사용자가 콘텐츠에 대해 HTTPS 요청을 전송할 경우 다음과 같은 작업이 수행됩니다.

1. 최종 사용자는 요청 URL에서 도메인 이름을 자동으로 가져와서 요청 헤더의 필드에 추가합니다.
2. CloudFront는 요청을 받으면 요청 헤더에서 도메인 이름을 찾아, 해당하는 SSL/TLS 인증서로 요청에 응답합니다.
3. 최종 사용자와 CloudFront는 SSL/TLS 협상을 수행합니다.
4. 요청된 콘텐츠를 CloudFront가 최종 사용자에 반환합니다.

SNI를 지원하는 최신 브라우저 목록은 Wikipedia 항목인 Server Name Indication을 참조하십시오.

SNI를 사용하고자 하지만 사용자 브라우저 중 일부에서 SNI가 지원되지 않는 경우 몇 가지 옵션 중 하나를 사용할 수 있습니다.

- SNI 대신에 전용 IP 주소를 사용하여 HTTPS 요청을 제공하도록 CloudFront를 구성합니다. 자세한 내용은 전용 IP 주소를 사용하여 HTTPS 요청 제공(모든 클라이언트에 적용) (p. 103) 단원을 참조하십시오.
- 사용자 지정 인증서 대신에 CloudFront SSL/TLS 인증서를 사용합니다. 그러기 위해서는 파일에 대한 URL에서 배포의 CloudFront 도메인 이름을 사용해야 합니다. 예: `https://d111111abcdef8.cloudfront.net/logo.png`.

  기본 CloudFront 인증서를 사용할 경우 최종 사용자가 SSL 프로토콜 TLSv1 또는 그 이상 버전을 지원해야 합니다. CloudFront는 기본 CloudFront 인증서로 SSLv3을 지원하지 않습니다.

  또한 CloudFront에서 사용 중인 SSL/TLS 인증서를 사용자 지정 인증서에서 기본 CloudFront 인증서로 다음과 같이 변경해야 합니다.
  - 배포를 사용하여 콘텐츠를 배포하지 않은 경우에는 구성을 변경하면 됩니다. 자세한 내용은 배포 업데이트 (p. 59) 단원을 참조하십시오.
  - 배포를 사용하여 콘텐츠를 배포한 경우에는 새 CloudFront 배포를 생성하고 파일에 대한 URL을 변경하여 콘텐츠를 사용할 수 없는 시간을 줄이거나 없애야 합니다. 자세한 내용은 사용자 지정 SSL/TLS 인증서에서 기본 CloudFront 인증서로 되돌리기 (p. 114) 단원을 참조하십시오.
- 사용자가 사용하는 브라우저를 개발자가 제어할 수 있는 경우 사용자가 해당 브라우저를 SNI가 지원되는 브라우저로 업그레이드하게 합니다.
- HTTPS 대신에 HTTP를 사용합니다.

전용 IP 주소를 사용하여 HTTPS 요청 제공(모든 클라이언트에 적용)

서버 이름 표시(SNI)는 요청을 도메인과 연결하는 한 가지 방법입니다. 또 다른 방법으로는 전용 IP 주소 사용을 들 수 있습니다. 2010년 이후 출시 브라우저나 클라이언트로 업그레이드가 불가한 사용자가 있다면 전

용 IP 주소를 사용하여 HTTPS 요청을 제공할 수 있습니다. SNI를 지원하는 최신 브라우저 목록은 Wikipedia 항목인 Server Name Indication을 참조하십시오.

> **Important**
>
> 전용 IP 주소를 사용하여 HTTPS 요청을 제공하도록 CloudFront를 구성하면 추가 월별 요금이 발생합니다. 요금 발생은 SSL/TLS 인증서를 배포와 연결하고 배포를 활성화할 때 시작됩니다. CloudFront 요금에 대한 자세한 내용은 Amazon CloudFront 요금을 참조하십시오. 또한 여러 개의 CloudFront 배포에 동일한 인증서 사용 (p. 108) 단원을 참조하십시오.

전용 IP 주소를 사용하여 HTTPS 요청을 제공하도록 CloudFront를 구성한 경우 CloudFront에서는 대체 도메인 이름을 각 CloudFront 엣지 로케이션의 전용 IP 주소와 연결합니다. 최종 사용자가 콘텐츠에 대해 HTTPS 요청을 전송할 경우 다음과 같은 작업이 수행됩니다.

1. DNS에서 배포의 해당 엣지 로케이션에 대한 IP 주소로 요청을 라우팅합니다.
2. CloudFront에서는 이 IP 주소를 사용하여 배포를 식별하고 최종 사용자에게 반환할 SSL/TLS 인증서를 결정합니다.
3. 최종 사용자와 CloudFront가 SSL/TLS 인증서를 사용하여 SSL/TLS 협상을 수행합니다.
4. 요청된 콘텐츠를 CloudFront가 최종 사용자에 반환합니다.

이 방법은 사용자가 사용 중인 브라우저나 기타 최종 사용자와 상관없이 모든 HTTPS 요청에 작동합니다.

CloudFront에서 SSL/TLS 인증서를 사용하기 위한 요구 사항

SSL/TLS 인증서에 대한 요구 사항은 이 주제에 설명되어 있습니다. 별도로 명시되지 않는 한 다음 두 가지를 모두 적용합니다.

- 최종 사용자와 CloudFront 사이에서 HTTPS를 사용한 인증서
- CloudFront와 오리진 사이에서 HTTPS를 사용한 인증서

주제

- 인증서 발행자 (p. 104)
- 인증서를 요청하는 AWS 리전(AWS Certificate Manager) (p. 105)
- 인증서 형식 (p. 105)
- 중간 인증서 (p. 105)
- 키 유형 (p. 106)
- 프라이빗 키 (p. 106)
- 권한 (p. 106)
- 퍼블릭 키의 크기 (p. 106)
- 지원되는 인증서 유형 (p. 106)
- 인증서 만료 날짜 및 갱신 (p. 107)
- CloudFront 배포 및 인증서의 도메인 이름 (p. 107)
- Minimum SSL Protocol Version (p. 107)
- Supported HTTP Versions (p. 107)

인증서 발행자

사용해야 하는 인증서 발행자는 최종 사용자와 CloudFront 간에 HTTPS를 요구할지 또는 CloudFront와 오리진 간에 HTTPS를 요구할지에 따라 다릅니다.

- 최종 사용자와 CloudFront 간의 HTTPS – 신뢰할 수 있는 인증 기관(CA)(예: Comodo, DigiCert, Symantec 등)에서 발급한 인증서를 사용하거나, AWS Certificate Manager(ACM)에서 제공한 인증서를 사용할 수 있습니다.

  Important

  CloudFront 배포에 대체 도메인 이름을 사용하려면, 대체 도메인 이름을 사용할 권한이 있음을 CloudFront에 확인해야 합니다. 이를 수행하려면 배포에 유효한 인증서를 연결하고 해당 인증서가 Mozilla에 포함된 CA 인증서 목록에 있는 신뢰할 수 있는 CA에서 제공한 것인지 확인해야 합니다. CloudFront는 자체 서명된 인증서를 사용하여 대체 도메인 이름을 사용할 권한을 확인하는 것을 허용하지 않습니다.

- CloudFront와 사용자 지정 오리진 간의 HTTPS – 오리진이 ELB 로드 밸런서(예: Amazon EC2)가 아닐 경우 인증서는 신뢰할 수 있는 CA(예: Comodo, DigiCert, Symantec 등)에서 발급한 것이어야 합니다. 오리진이 ELB 로드 밸런서일 경우 ACM에서 제공한 인증서를 사용할 수도 있습니다.

  Important

  CloudFront에서 HTTPS를 사용하여 오리진과 통신할 경우, CloudFront에서는 인증서가 신뢰할 수 있는 인증 기관에 의해 발급되었는지 확인합니다. CloudFront에서는 Mozilla가 지원하는 것과 동일한 인증 기관을 지원합니다. 최신 목록은 Mozilla Included CA Certificate List를 참조하십시오. CloudFront와 오리진 간의 HTTPS 통신에는 자체 서명한 인증서를 사용할 수 없습니다.

SSL/TLS 인증서 가져오기 및 설치에 대한 자세한 내용은 HTTP 서버 소프트웨어 설명서와 인증 기관 설명서를 참조하십시오. ACM에 대한 자세한 내용은 AWS Certificate Manager 사용 설명서를 참조하십시오.

인증서를 요청하는 AWS 리전(AWS Certificate Manager)

최종 사용자와 CloudFront 간에 HTTPS를 요구하려면, 인증서를 요청하거나 가져오기 전에 먼저 AWS Certificate Manager 콘솔에서 AWS 리전을 미국 동부(버지니아 북부)로 변경해야 합니다.

CloudFront와 오리진 간에 HTTPS를 요구할 경우, 오리진으로 ELB 로드 밸런서를 사용한다면 아무 리전에서 인증서를 요청하거나 가져올 수 있습니다.

인증서 형식

인증서는 X.509 PEM 형식이어야 합니다. AWS Certificate Manager을 사용할 경우 이 형식이 기본 형식입니다.

중간 인증서

다른 인증 기관(CA)을 사용할 경우, 도메인 인증서에 서명한 CA 관련 인증서부터, 인증서 체인에 있는 모든 중간 인증서를 .pem 파일에 명시합니다. 일반적으로 올바른 체인 순서대로 중간 및 루트 인증서를 나열하는 파일이 CA 웹 사이트에 있습니다.

Important

루트 인증서, 신뢰 경로에 없는 중간 인증서 또는 CA의 퍼블릭 키 인증서는 포함하지 마세요.

다음은 그 예입니다:

```
-----BEGIN CERTIFICATE-----
Intermediate certificate 2
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
Intermediate certificate 1
-----END CERTIFICATE-----
```

키 유형

CloudFront는 RSA 퍼블릭/프라이빗 키 페어만 지원합니다.

프라이빗 키

다른 인증 기관(CA)의 인증서를 사용할 경우 다음 사항에 유의하세요.

- 프라이빗 키는 인증서에 있는 퍼블릭 키와 일치해야 합니다.
- 프라이빗 키는 PEM 형식의 RSA 프라이빗 키여야 합니다. PEM 머리글은 `BEGIN RSA PRIVATE KEY`이고, 바닥글은 `END RSA PRIVATE KEY`입니다.
- 프라이빗 키는 암호로 암호화할 수 없습니다.

AWS Certificate Manager(ACM)에서 인증서를 제공할 경우 ACM은 프라이빗 키를 공개하지 않습니다. 이 프라이빗 키는 ACM과 통합된 AWS 서비스에서 사용하기 위해 ACM에 저장됩니다.

권한

인증서를 콘텐츠 전송 네트워크(CDN)로 가져오려면 인증서를 발급한 인증 기관(CA)의 사용 권한을 포함하여 SSL/TLS 인증서를 사용하고 가져올 수 있는 권한이 있어야 합니다.

AWS Certificate Manager(ACM)를 사용하는 경우, AWS Identity and Access Management 권한을 사용하여 인증서에 대한 액세스를 제한하는 것이 좋습니다. 자세한 내용은 AWS Certificate Manager 사용 설명서에서 권한 및 정책을 참조하십시오.

퍼블릭 키의 크기

인증서에 대한 퍼블릭 키 길이는 저장 위치에 따라 다릅니다.

- AWS Certificate Manager(ACM)으로 인증서 가져오기: 퍼블릭 키 길이는 1024비트 또는 2048비트여야 합니다. CloudFront에서는 더 큰 키가 지원되기는 하지만, ACM에서 사용하는 인증서의 최대 길이는 2048비트입니다.
- AWS Identity and Access Management(IAM) 인증서 스토어에 인증서 업로드: 퍼블릭 키의 최대 크기는 2048비트입니다.

2048비트를 사용하는 것이 좋습니다.

ACM에서 제공한 인증서의 퍼블릭 키에 대한 자세한 내용은 AWS Certificate Manager 사용 설명서의 ACM 인증서 특성을 참조하십시오.

퍼블릭 키 크기를 확인하는 방법에 대한 자세한 내용 SSL/TLS 인증서에서 퍼블릭 키 크기 확인 (p. 111)을 참조하십시오.

지원되는 인증서 유형

CloudFront는 다음을 포함한 모든 유형의 인증서를 지원합니다.

- 도메인 검증 인증서
- 확장 검증(EV) 인증서
- 높은 보증 인증서
- 와일드카드 인증서(`*.example.com`)
- 제목 대체 이름(SAN) 인증서(`example.com` 및 `example.net`)

인증서 만료 날짜 및 갱신

다른 인증 기관(CA)에서 받은 인증서를 사용할 경우, 인증서 만료 날짜를 직접 모니터링해야 하고, AWS Certificate Manager(ACM)으로 가져오거나 AWS Identity and Access Management 인증서 스토어에 업로드한 SSL/TLS 인증서를 직접 갱신해야 합니다.

ACM에서 제공한 인증서를 사용할 경우 ACM에서 인증서 갱신을 관리해 줍니다. 자세한 내용은 AWS Certificate Manager 사용 설명서의 관리형 갱신을 참조하십시오.

CloudFront 배포 및 인증서의 도메인 이름

사용자 지정 오리진을 사용하는 경우 오리진의 SSL/TLS 인증서에는 [Common Name] 필드에 도메인 이름이 포함되며 Subject Alternative Names(주제 대체 이름) 필드에도 몇 번 더 포함될 수 있습니다. CloudFront는 인증서 도메인 이름 내 와일드카드 문자 사용을 지원합니다.

> **Important**
>
> 배포에 대체 도메인 이름을 추가하면, CloudFront는 해당 대체 도메인 이름이 연결한 인증서에 포함되어 있는지 확인합니다. 인증서는 인증서의 주체 대체 이름(SAN) 필드에 있는 대체 도메인 이름을 포함해야 합니다. 즉 해당 도메인 이름과 정확히 일치하거나 추가할 대체 도메인 이름과 동일한 수준에서 와일드카드를 포함해야 합니다.
> 자세한 내용은 대체 도메인 이름 사용과 관련된 요구 사항 (p. 73) 단원을 참조하십시오.

인증서의 도메인 이름 중 하나는 오리진 도메인 이름으로 지정하는 도메인 이름과 일치해야 합니다. 일치하는 도메인 이름이 없는 경우 CloudFront는 최종 사용자에게 HTTP 상태 코드 502(잘못된 게이트웨이)를 반환합니다.

Minimum SSL Protocol Version

전용 IP 주소를 사용할 경우 보안 정책을 선택하여 최종 사용자와 CloudFront 간의 연결을 위한 최소 SSL 프로토콜 버전을 선택할 수 있습니다.

자세한 내용은 보안 정책 (p. 53) 주제에서 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.

Supported HTTP Versions

한 인증서를 둘 이상의 CloudFront 배포와 연결하는 경우, 인증서와 연결된 모든 배포에서 Supported HTTP Versions (p. 54)에 대해 동일한 옵션을 사용해야 합니다. CloudFront 배포를 생성 또는 업데이트할 때 이 옵션을 지정합니다.

CloudFront에서 SSL/TLS 인증서 사용 시의 할당량(최종 사용자와 CloudFront 간의 HTTPS만 해당)

CloudFront에서 SSL/TLS 인증서 사용 시 다음 할당량(이전에는 제한이라고 함)에 유의하십시오. 이러한 할당량은 최종 사용자와 CloudFront 간의 HTTPS 통신을 위해 AWS Certificate Manager(ACM)을 사용하여 제공하거나, ACM으로 가져오거나, IAM 인증서 스토어에 업로드한 SSL/TLS 인증서에만 적용됩니다.

CloudFront 배포당 최대 인증서 수

최대 한 개의 SSL/TLS 인증서를 CloudFront 배포와 연결할 수 있습니다.

ACM으로 가져오거나 IAM 인증서 스토어로 업로드할 수 있는 인증서의 최대 수

다른 인증 기관에서 SSL/TLS 인증서를 구입한 경우 다음 위치 중 하나에 인증서를 저장해야 합니다.

- AWS Certificate Manager – ACM 인증서 수의 현재 할당량은 AWS Certificate Manager 사용 설명서의 할당량을 참조하십시오. 명시된 할당량은 ACM을 사용하여 제공한 인증서와 ACM으로 가져온 인증서를 모두 포함한 총 개수입니다.

- IAM 인증서 스토어 – 하나의 AWS 계정에서 IAM 인증서 스토어에 업로드할 수 있는 인증서 수의 현재 할당량(이전에는 제한이라고 함)은 IAM 사용 설명서의 IAM 및 STS 제한을 참조하십시오. AWS Management 콘솔에서 더 높은 할당량을 요청할 수 있습니다.

AWS 계정당 인증서의 최대 수(전용 IP 주소만 해당)

전용 IP 주소를 사용하여 HTTPS 요청을 처리하려는 경우 다음 사항에 유의하세요.

- 기본적으로 CloudFront는 AWS 계정에 두 인증서를 사용할 수 있는 권한(일일 사용을 위한 인증서와 여러 배포용으로 인증서를 교체해야 할 때 사용할 인증서)을 부여합니다.
- 이미 이 기능을 사용하고 있지만 AWS 계정에서 사용할 수 있는 사용자 지정 SSL/TLS 인증서 수를 늘려야 하는 경우에는 지원 센터로 이동하여 사례를 생성하십시오. 사용 권한이 필요한 인증서 수를 지정하고 요청 상황을 설명하세요. 그러고 나면 귀하의 계정을 가능한 빨리 업데이트하겠습니다.

다른 AWS 계정을 사용하여 생성된 CloudFront 배포에 동일한 인증서 사용

다른 인증 기관을 사용할 경우 그리고 서로 다른 AWS 계정으로 생성된 여러 CloudFront 배포에서 동일한 인증서를 사용하려면 각 AWS 계정마다 한 번씩 해당 인증서를 ACM으로 가져오거나 IAM 인증서 스토어로 업로드해야 합니다.

ACM에서 제공한 인증서를 사용하는 경우, 다른 AWS 계정이 생성한 인증서를 사용하도록 CloudFront를 구성할 수 없습니다.

CloudFront 및 다른 AWS 서비스에 동일한 인증서 사용(IAM 인증서 스토어만 해당)

신뢰할 수 있는 인증 기관(예: Comodo, DigiCert, Symantec 등)에서 인증서를 구입한 경우, CloudFront 및 기타 AWS 서비스에서 동일한 인증서를 사용할 수 있습니다. ACM으로 인증서를 가져올 경우 한 번만 가져와서 여러 AWS 서비스에 사용해야 합니다.

ACM에서 제공한 인증서를 사용할 경우 인증서가 ACM에 저장됩니다.

여러 개의 CloudFront 배포에 동일한 인증서 사용

HTTPS 요청을 처리하는 데 사용하는 임의의 또는 모든 CloudFront 배포에서 동일한 인증서를 사용할 수 있습니다. 다음을 참조하십시오.

- 전용 IP 주소를 사용한 요청 제공과 SNI를 사용한 요청 제공에 동일한 인증서를 사용할 수 있습니다.
- 단 한 개의 인증서를 각각의 배포와 연결할 수 있습니다.
- 각각의 배포에 하나 이상의 대체 도메인 이름이 포함되어야 하며, 이 이름은 인증서의 Common Name 필드나 Subject Alternative Names 필드에도 표시됩니다.
- 전용 IP 주소를 사용하여 HTTPS 요청을 제공하고 동일한 AWS 계정을 사용해 모든 배포를 생성한 경우, 모든 배포에 대해 동일한 인증서가 사용되므로 비용을 크게 절감할 수 있습니다. CloudFront는 배포가 아닌 인증서별로 요금을 청구합니다.

  예를 들어, 동일한 AWS 계정을 사용하여 세 개의 배포를 생성하고, 이 세 개의 배포에 모두 동일한 인증서를 사용한다고 가정하겠습니다. 이 경우, 전용 IP 주소 사용에 대한 단 한 건의 수수료만 청구됩니다.

  하지만 전용 IP 주소를 사용하여 HTTPS 요청을 제공하고 서로 다른 AWS 계정에서 동일한 인증서를 사용해 CloudFront 배포를 생성할 경우, 각각의 계정에 전용 IP 주소 사용에 대한 수수료가 청구됩니다. 예를 들어, 세 개의 AWS 계정을 사용하여 세 개의 배포를 생성하고 이 세 배포에 동일한 인증서를 사용할 경우, 각 계정에 전용 IP 주소 사용에 대한 수수료 전액이 청구됩니다.

대체 도메인 이름과 HTTPS 구성

파일의 URL에 대체 도메인 이름을 사용하고, 최종 사용자와 CloudFront 간에 HTTPS를 사용하려면 해당 절차를 수행합니다.

주제

- 3개 이상의 SSL/TLS 인증서를 사용할 수 있는 권한 요청 (p. 109)

- SSL/TLS 인증서 받기 (p. 109)
- SSL/TLS 인증서 가져오기 (p. 109)
- CloudFront 배포 업데이트 (p. 110)

3개 이상의 SSL/TLS 인증서를 사용할 수 있는 권한 요청

CloudFront에 3개 이상의 SSL/TLS 전용 IP 인증서를 영구적으로 연결할 수 있는 권한이 필요할 경우 다음과 같이 합니다. HTTPS 요청에 대한 자세한 내용은 CloudFront에서 HTTPS 요청을 제공하는 방식 선택 (p. 102) 단원을 참조하십시오.

> **Note**
>
> 이 절차는 CloudFront 배포에 3개 이상의 DedicatedIP 인증서를 사용하는 경우에 사용됩니다. 기본값은 2입니다. 배포에 2개 이상의 SSL 인증서를 바인딩할 수 없다는 점에 유의하세요.
> 한 번에 하나의 SSL/TLS 인증서만 CloudFront 배포에 연결할 수 있습니다. 이 수치는 모든 CloudFront 배포에서 사용할 수 있는 전용 IP SSL 인증서의 총 개수에 해당됩니다.

CloudFront 배포에 3개 이상의 인증서를 사용할 수 있는 권한을 요청하려면

1. 지원 센터로 이동하여 사례를 생성합니다.
2. 사용 권한이 필요한 인증서 수를 지정하고 요청 상황을 설명하세요. 그러고 나면 귀하의 계정을 가능한 빨리 업데이트하겠습니다.
3. 다음 절차로 진행합니다.

SSL/TLS 인증서 받기

SSL/TLS 인증서를 받습니다(아직 없는 경우). 자세한 내용은 다음과 같이 해당 문서를 참조하십시오.

- AWS Certificate Manager(ACM)에서 제공한 인증서를 사용하려면 AWS Certificate Manager 사용 설명서를 참조하십시오. 그런 다음 CloudFront 배포 업데이트 (p. 110) 단원으로 건너뛰십시오.

  > **Note**
  >
  > AWS 관리형 리소스에 대해 SSL/TLS 인증서를 프로비저닝 및 관리하고 배포할 때 ACM을 사용하는 것이 좋습니다.

- 다른 인증 기관(CA)의 인증서를 받으려면 인증 기관에서 제공한 설명서를 참조하십시오. 인증서가 있으면 다음 절차로 진행합니다.
- 자체 서명한 인증서를 만들려면 인증서를 만들고 서명하는 데 사용할 애플리케이션의 설명서를 참조하십시오. 그리고 다음 절차로 진행합니다.

SSL/TLS 인증서 가져오기

다른 인증 기관의 인증서를 받았을 경우 ACM으로 가져오거나 IAM 인증서 스토어에 업로드합니다.

ACM(권장)

ACM에서 ACM 콘솔을 사용하거나 프로그래밍 방식으로 타사 인증서를 가져올 수 있습니다. 인증서를 ACM로 가져오는 작업에 대한 자세한 내용은 AWS Certificate Manager 사용 설명서의 AWS Certificate Manager로 인증서 가져오기 단원을 참조하십시오.

IAM 인증서 스토어

해당 리전에서 ACM을 사용할 수 없으면 다음 AWS CLI 명령을 사용하여 타사 인증서를 IAM 인증서 스토어에 업로드합니다. (ACM을 사용할 수 있는 리전 목록을 보려면 Amazon Web Services 일반 참조의 "AWS 리전 및 엔드포인트" 장에서 AWS Certificate Manager를 참조하십시오.)

```
aws iam upload-server-certificate --server-certificate-name CertificateName --
certificate-body file://public_key_certificate_file --private-key file://privatekey.pem
 --certificate-chain file://certificate_chain_file --path /cloudfront/path/
```

다음을 참조하십시오.

- AWS 계정 – CloudFront 배포를 생성하는 데 사용한 것과 동일한 AWS 계정을 사용하여 인증서를 IAM 인증서 스토어로 업로드해야 합니다.
- --경로 파라미터 – 인증서를 IAM으로 업로드할 때 `-path` 파라미터(인증서 경로) 값은 `/cloudfront/`로 시작해야 합니다(`/cloudfront/production/` 또는 `/cloudfront/test/`). 경로는 /로 끝나야 합니다.
- Existing certificates – `--server-certificate-name` 및 `--path` 파라미터에, 기존 인증서와 연결되어 있는 값과 다른 값을 지정해야 합니다.
- CloudFront 콘솔 사용 – AWS CLI에서 `--server-certificate-name` 파라미터에 지정하는 값(예: `myServerCertificate`)은 CloudFront 콘솔의 SSL 인증서 목록에 나타납니다.
- CloudFront API 사용 – AWS CLI에서 반환하는 영숫자 문자열(예: `AS1A2M3P4L5E67SIIXR3J`)을 메모해 둡니다. 이 문자열은 `IAMCertificateId` 요소에서 지정하는 값입니다. IAM ARN은 필요하지 않습니다. CLI에서도 반환되기 때문입니다.

AWS CLI에 대한 자세한 내용은 AWS Command Line Interface 사용 설명서 및 AWS CLI Command Reference 단원을 참조하십시오.

CloudFront 배포 업데이트

배포의 설정을 업데이트하려면 다음과 같이 합니다.

대체 도메인 이름을 사용하도록 CloudFront 배포를 구성하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 업데이트하려는 배포의 ID를 선택합니다.
3. [General] 탭에서 [Edit]를 선택합니다.
4. 다음 값을 업데이트합니다.

   Alternate Domain Names (CNAMEs)

   해당 대체 도메인 이름을 추가합니다. 도메인 이름을 쉼표로 구분하거나 각각의 이름을 새 줄에 입력합니다.

   SSL 인증서(웹 배포만 해당)

   Custom SSL Certificate(사용자 지정 SSL 인증서)를 선택하고 목록에서 인증서를 선택합니다.

   최대 100개의 인증서가 여기에 나열됩니다. 100개 이상의 인증서가 있고 추가할 인증서가 보이지 않는 경우에는 해당 필드에 인증서 ARN을 입력하여 선택할 수 있습니다.

   IAM 인증서 스토어에 인증서를 업로드했는데 목록에 나타나지 않아서 필드에 이름을 입력하는 방법으로 선택을 할 수 없는 경우에는 SSL/TLS 인증서 가져오기 (p. 109) 절차를 검토하여 인증서를 올바르게 업로드했는지 확인합니다.

   > **Important**
   >
   > SSL/TLS 인증서를 CloudFront 배포와 연결한 후에 모든 배포에서 인증서를 제거하고 배포의 상태가 배포 완료로 변경될 때까지 ACM 또는 IAM 인증서 스토어에서 인증서를 삭제하지 마십시오.

클라이언트 지원(웹 배포만 해당)

다음 중 해당 옵션을 선택합니다.

- All Clients(모든 클라이언트): CloudFront가 전용 IP 주소를 사용하여 HTTPS 콘텐츠를 제공합니다. 이 옵션을 선택하면 SSL/TLS 인증서를 활성화된 배포와 연결할 때 추가 요금이 발생합니다. 자세한 내용은 Amazon CloudFront 요금을 참조하십시오.
- Only Clients that Support Server Name Indication (SNI)(서버 이름 표시(SNI)를 지원하는 클라이언트만): SNI를 지원하지 않는 이전 브라우저 또는 기타 클라이언트는 다른 방법을 사용하여 콘텐츠에 액세스해야 합니다.

자세한 내용은 CloudFront에서 HTTPS 요청을 제공하는 방식 선택 (p. 102) 단원을 참조하십시오.

5. 예, 편집합니다를 선택합니다.
6. CloudFront에서 최종 사용자와 CloudFront 간에 HTTPS를 요구하도록 구성:
   a. 동작 탭에서 업데이트할 캐시 동작을 선택하고 편집을 선택합니다.
   b. Viewer Protocol Policy(최종 사용자 프로토콜 정책)에 다음 값 중 하나를 지정합니다.

      Redirect HTTP to HTTPS

      최종 사용자가 두 프로토콜 모두 사용할 수 있지만 HTTP 요청은 HTTPS 요청으로 자동 리디렉션됩니다. CloudFront는 HTTP 상태 코드 301(영구 이동됨)을 새로운 HTTPS URL과 함께 반환합니다. 그러면 최종 사용자는 HTTPS URL을 사용하여 이 요청을 CloudFront에 다시 제출합니다.

      Important

      CloudFront는 HTTP에서 HTTPS로 `DELETE`, `OPTIONS`, `PATCH`, `POST` 또는 `PUT` 요청을 리디렉션하지 않습니다. HTTPS에 리디렉션할 캐시 동작을 구성하는 경우, CloudFront는 해당 캐시 동작에 대한 HTTP `DELETE`, `OPTIONS`, `PATCH`, `POST` 또는 `PUT` 요청에 HTTP 상태 코드 403(금지됨)으로 응답합니다.

      최종 사용자가 HTTPS 요청으로 리디렉션되는 HTTP 요청을 만들 경우 CloudFront에서 두 요청 모두에 대해 요금을 부과합니다. HTTP 요청의 경우에는 CloudFront에서 최종 사용자에게 반환되는 요청 및 헤더에만 요금이 부과됩니다. HTTPS 요청의 경우에는 요청, 헤더, 그리고 오리진에 의해 반환된 파일을 모두 처리합니다.

      HTTPS Only

      최종 사용자가 HTTPS를 사용할 경우에만 콘텐츠에 액세스할 수 있습니다. 최종 사용자가 HTTPS 요청 대신에 HTTP 요청을 보내면 CloudFront은(는) HTTP 상태 코드 403(금지됨)을 반환하고 파일은 반환하지 않습니다.
   c. 예, 편집합니다를 선택합니다.
   d. 최종 사용자와 CloudFront 간에 HTTPS를 요구하려는 캐시 동작 각각에 대해 단계 c까지 전체 단계를 반복합니다.
7. 업데이트한 구성을 프로덕션 환경에서 사용하기 전에 다음 사항을 확인합니다.
   - 각 캐시 동작의 경로 패턴이 최종 사용자에 HTTPS를 사용하도록 지정한 요청에만 적용되는가.
   - 캐시 동작이 CloudFront에서 평가하도록 할 순서대로 나열되었는가. 자세한 내용은 경로 패턴 (p. 44) 단원을 참조하십시오.
   - 캐시 동작이 올바른 오리진에 요청을 라우팅하는가.

SSL/TLS 인증서에서 퍼블릭 키 크기 확인

CloudFront 대체 도메인 이름 및 HTTPS를 사용하는 경우, SSL/TLS 인증서의 퍼블릭 키의 최대 크기는 2048 비트입니다. (이는 퍼블릭 키의 문자 수가 아닌 비트입니다.) 인증서에서 AWS Certificate Manager를 사

용하는 경우에는 ACM이 더 큰 키를 지원하더라도 CloudFront에서 더 큰 키를 사용할 수 없다는 점에 유의하십시오.

다음 OpenSSL 명령을 실행하여 퍼블릭 키의 크기를 확인할 수 있습니다.

```
openssl x509 -in path and filename of SSL/TLS certificate -text -noout
```

여기서 각 항목은 다음과 같습니다.

- `-in`은 SSL/TLS 인증서의 경로 및 파일 이름을 지정합니다.
- `-text`를 지정하면 OpenSSL에서 퍼블릭 키 길이를 비트로 표시합니다.
- `-noout`를 지정하면 OpenSSL에서 퍼블릭 키가 표시되지 않습니다.

출력 예:

```
Public-Key: (2048 bit)
```

SSL/TLS 인증서에 대한 할당량 증가

AWS Certificate Manager로 가져오거나 AWS Identity and Access Management에 업로드할 수 있는 SSL/TLS 인증서 수에는 할당량(이전에는 제한이라고 함)이 있습니다. 또한 전용 IP 주소를 사용하여 HTTPS 요청을 제공하도록 CloudFront를 구성할 경우 AWS 계정에서 사용할 수 있는 SSL/TLS 인증서 개수에도 할당량이 있습니다. 하지만 더 높은 할당량을 요청할 수 있습니다.

주제

- ACM로 가져올 수 있는 인증서 (p. 112)
- IAM에 업로드할 수 있는 인증서 (p. 112)
- 전용 IP 주소와 함께 사용할 수 있는 인증서 (p. 113)

ACM로 가져올 수 있는 인증서

ACM으로 가져올 수 있는 인증서 개수에 대한 할당량은 AWS Certificate Manager 사용 설명서의 할당량을 참조하십시오.

할당량을 더 높여 달라고 요청하려면 AWS 지원 센터에서 사례를 생성합니다. 다음 값을 지정합니다.

- Service Limit Increase(서비스 제한 증가)의 기본값을 적용합니다.
- Limit type(제한 유형)에서 Certificate Manager를 선택합니다.
- Region(리전)에서 인증서를 가져올 AWS 리전을 선택합니다.
- Limit(제한)에서 Number of ACM Certificates(ACM 인증서 개수)를 선택합니다.

그런 다음 양식을 마저 작성하여 제출합니다.

IAM에 업로드할 수 있는 인증서

IAM에 업로드할 수 있는 인증서 개수에 대한 할당량(이전에는 제한이라고 함)은 IAM 사용 설명서의 IAM 및 STS 제한을 참조하십시오.

할당량을 더 높여 달라고 요청하려면 AWS 지원 센터에서 사례를 생성합니다. 다음 값을 지정합니다.

- Service Limit Increase(서비스 제한 증가)의 기본값을 적용합니다.
- Limit type(제한 유형)에서 Certificate Manager를 선택합니다.
- Region(리전)에서 인증서를 가져올 AWS 리전을 선택합니다.

- Limit(제한)에서 Server Certificate Limit(IAM)(서버 인증서 제한(IAM))을 선택합니다.

그런 다음 양식을 마저 작성하여 제출합니다.

전용 IP 주소와 함께 사용할 수 있는 인증서

전용 IP 주소를 사용하여 HTTPS 요청을 제공할 경우 각 AWS 계정에서 사용할 수 있는 SSL 인증서 개수에 대한 할당량(이전에는 제한이라고 함)은 SSL 인증서에 대한 할당량(웹 배포만 해당) (p. 449) 단원을 참조하십시오.

할당량을 더 높여 달라고 요청하려면 AWS 지원 센터에서 사례를 생성합니다. 다음 값을 지정합니다.

- Service Limit Increase(서비스 제한 증가)의 기본값을 적용합니다.
- Limit Type(제한 유형)에서 CloudFront Distributions(CloudFront 배포)를 선택합니다.
- Limit(제한)에서 Dedicated IP SSL Certificate Limit per Account(계정당 전용 IP SSL 인증서 제한)를 선택합니다.

그런 다음 양식을 마저 작성하여 제출합니다.

SSL/TLS 인증서 회전

AWS Certificate Manager(ACM)에서 제공한 인증서를 사용하는 경우, SSL/TLS 인증서를 교체할 필요가 없습니다. ACM이 사용자 대신 인증서 갱신을 관리합니다. 자세한 내용은 AWS Certificate Manager 사용 설명서의 관리형 갱신을 참조하십시오.

> **Note**
>
> ACM은 다른 인증 기관에서 구입하여 ACM으로 가져온 인증서에 대해서는 갱신을 관리하지 않습니다.

다른 인증 기관의 인증서를ACM으로 가져오거나(권장) IAM 인증서 스토어에 업로드한 경우 인증서를 교체해야 할 경우가 있습니다. 예를 들어, 인증서 만료 날짜가 다가올 경우 인증서를 교체해야 합니다.

> **Important**
>
> 전용 IP 주소를 사용하여 HTTPS 요청을 제공하도록 CloudFront를 구성한 경우 인증서를 교체하는 동안은 하나 이상의 추가 인증서 사용에 대한 비례 할당으로 계산된 추가 요금이 발생할 수 있습니다. 추가 요금을 최소화하기 위해 배포를 즉시 업데이트할 것을 권장합니다.

인증서를 교체하려면 다음 절차를 수행합니다. 최종 사용자는 인증서를 교체하는 동안은 물론, 프로세스가 완료된 후에도 콘텐츠에 계속 액세스할 수 있습니다.

SSL/TLS 인증서를 교체하려면

1. SSL/TLS 인증서에 대한 할당량 증가 (p. 112)는 추가 SSL 인증서를 사용하기 위한 권한이 필요한지 여부를 확인합니다. 권한이 필요한 경우 권한을 요청하고 기다렸다 권한이 부여되면 2단계로 넘어갑니다.
2. 새 인증서를 ACM으로 가져오거나 IAM에 업로드합니다. 자세한 내용은 Amazon CloudFront 개발자 안내서의 SSL/TLS 인증서 가져오기 단원을 참조하십시오.
3. 새 인증서를 사용하도록 한 번에 하나씩 배포를 업데이트합니다. 자세한 내용은 Amazon CloudFront 개발자 안내서의 CloudFront 배포 나열, 보기 및 업데이트 단원을 참조하십시오.
4. (선택 사항) 모든 CloudFront 배포를 업데이트한 후에는 ACM 또는 IAM에서 이전 인증서를 삭제할 수 있습니다.

   > **Important**
   >
   > 모든 배포에서 해당 인증서가 제거되고 업데이트된 배포의 상태가 `Deployed`로 변경될 때까지는 SSL/TLS 인증서를 삭제하지 마세요.

사용자 지정 SSL/TLS 인증서에서 기본 CloudFront 인증서로 되돌리기

CloudFront에서 최종 사용자와 CloudFront간에 HTTPS를 사용하도록 구성했으며 CloudFront에서 사용자 지정 SSL/TLS 인증서를 사용하도록 구성한 경우, 기본 CloudFront SSL/TLS 인증서를 사용하도록 구성을 변경할 수 있습니다. 이 절차는 콘텐츠를 배포하는 데 배포를 사용했는지 여부에 따라 다릅니다.

- 배포를 사용하여 콘텐츠를 배포하지 않은 경우에는 구성을 변경하면 됩니다. 자세한 내용은 배포 업데이트 (p. 59) 단원을 참조하십시오.
- 배포를 사용하여 콘텐츠를 배포한 경우에는 새 CloudFront 배포를 생성하고 파일에 대한 URL을 변경하여 콘텐츠를 사용할 수 없는 시간을 줄이거나 없애야 합니다. 이렇게 하려면 다음과 같이 합니다.

기본 CloudFront 인증서로 되돌리려면

1. 원하는 구성을 사용하여 새 CloudFront 배포를 생성합니다. SSL Certificate(SSL 인증서)에서 기본 CloudFront 인증서(*.cloudfront.net)를 선택합니다.

   자세한 정보는 배포 만들기 단계(개요) (p. 34) 단원을 참조하십시오.
2. CloudFront을(를) 사용하여 배포하고 있는 파일의 경우에는 CloudFront에서 새 배포에 배정한 도메인 이름이 사용되도록 애플리케이션에서 URL을 업데이트합니다. 예를 들면 `https://www.example.com/images/logo.png`를 `https://d111111abcdef8.cloudfront.net/images/logo.png`로 변경합니다.
3. 사용자 지정 SSL/TLS 인증서와 연결된 배포를 삭제하거나, SSL Certificate(SSL 인증서)의 값이 기본 CloudFront 인증서(*.cloudfront.net)로 변경되도록 배포를 업데이트합니다. 자세한 내용은 배포 업데이트 (p. 59) 단원을 참조하십시오.

   > **Important**
   >
   > 이 단계를 완료하지 않으면 AWS에서 사용자 지정 SSL/TLS 인증서 사용에 요금을 계속 부과합니다.
4. (선택 사항) 사용자 지정 SSL/TLS 인증서를 삭제합니다.

   a. AWS CLI 명령 `list-server-certificates`를 실행하여 삭제할 인증서의 인증서 ID를 확인합니다. 자세한 내용은 AWS CLI Command Reference의 list-server-certificates 단원을 참조하십시오.

   b. AWS CLI 명령 `delete-signing-certificate`를 실행하여 인증서를 삭제합니다. 자세한 내용은 AWS CLI Command Reference의 delete-signing-certificate 단원을 참조하십시오.

사용자 지정 SSL/TLS 인증서를 전용 IP 주소 사용에서 SNI 사용으로 전환

사용자 지정 SSL/TLS 인증서에 전용 IP 주소를 사용하도록 CloudFront를 구성한 경우 SSL/TLS 인증서에 SNI를 대신 사용하고 전용 IP 주소와 연결된 요금이 청구되지 않도록 전환할 수 있습니다. 다음 절차에서는 이 방법을 보여 줍니다.

> **Important**
>
> CloudFront 구성에 대한 이러한 업데이트는 SNI를 지원하는 최종 사용자에게는 아무런 효과가 없습니다. 이러한 최종 사용자는 변경 전/후에 콘텐츠에 액세스할 수 있을 뿐만 아니라, 변경이 CloudFront 엣지 로케이션에 전파되고 있는 동안에도 콘텐츠에 액세스할 수 있습니다. SNI를 지원하지 않는 최종 사용자는 변경 후에 콘텐츠에 액세스할 수 없습니다. 자세한 내용은 CloudFront에서 HTTPS 요청을 제공하는 방식 선택 (p. 102) 단원을 참조하십시오.

사용자 지정 SSL/TLS 인증서를 전용 IP 주소 사용에서 SNI 사용으로 전환하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 보거나 업데이트하려는 배포의 ID를 선택합니다.
3. Distribution Settings(배포 설정)를 선택합니다.
4. [General] 탭에서 [Edit]를 선택합니다.
5. Custom SSL Client Support(사용자 지정 SSL 클라이언트 지원)의 설정을 Only Clients that Support Server Name Indication(SNI)(SNI(서버 표시 이름)를 지원하는 클라이언트만)으로 변경합니다.
6. 예, 편집합니다를 선택합니다.

서명된 URL과 서명된 쿠키를 사용하여 프라이빗 콘텐츠 제공

인터넷을 통해 콘텐츠를 배포하는 많은 기업에서는 유료 사용자 등 일부 사용자용으로 제작된 각종 문서, 비즈니스 데이터, 미디어 스트림 또는 콘텐츠에 대한 액세스를 제한하고자 합니다. CloudFront를 통해 이러한 프라이빗 콘텐츠를 안전하게 제공하려면 다음과 같이 하십시오.

- 사용자가 특별한 CloudFront 서명된 URL 또는 서명된 쿠키를 사용하여 프라이빗 콘텐츠에 액세스하도록 합니다.
- 사용자가 오리진 서버에서 직접 콘텐츠에 액세스하는 URL(예: Amazon S3 또는 프라이빗 HTTP 서버)이 아닌 CloudFront URL을 사용하여 콘텐츠에 액세스해야 합니다. 반드시 CloudFront URL을 사용해야 하는 것은 아니지만, 서명된 URL 또는 서명된 쿠키에 지정한 제약 조건을 사용자가 우회하지 못하도록 하려면 사용하는 것이 좋습니다.

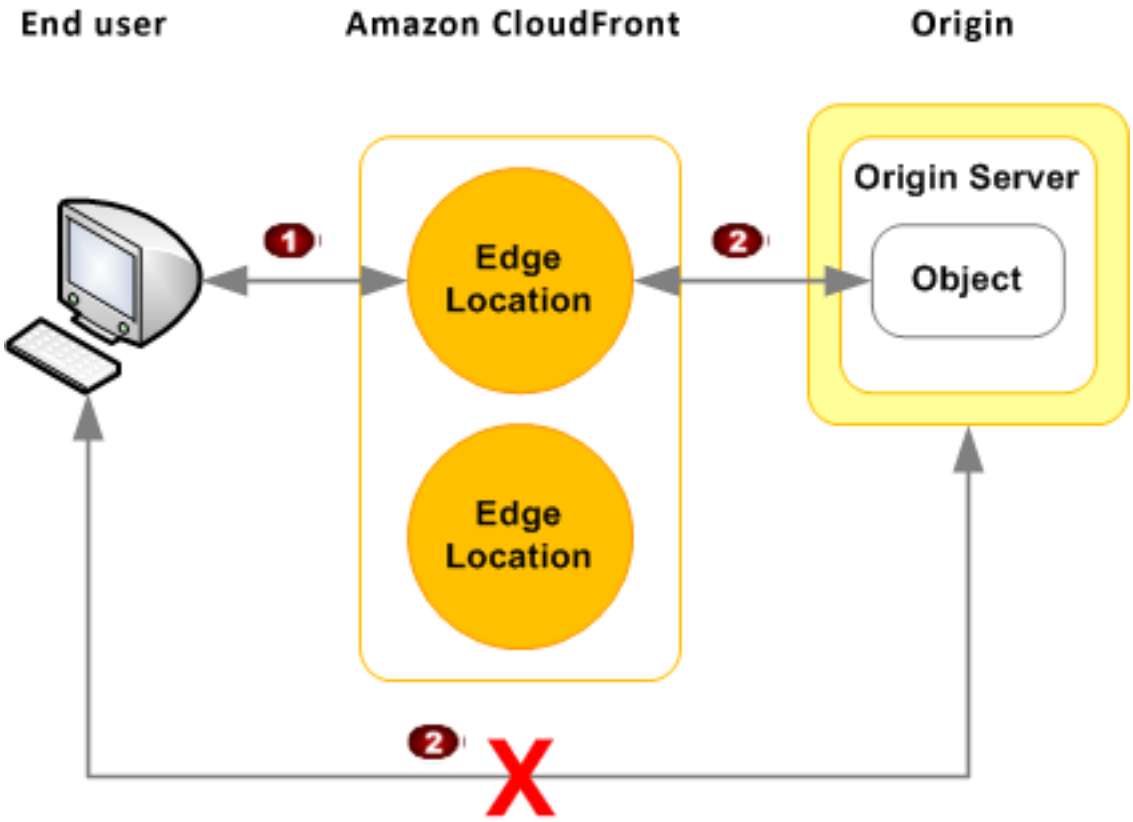
주제

- 프라이빗 콘텐츠 제공 개요 (p. 115)
- 작업 목록: 프라이빗 콘텐츠 제공 (p. 117)
- 서명된 URL 및 서명된 쿠키(신뢰할 수 있는 서명자)를 만들 수 있는 AWS 계정 지정 (p. 118)
- 서명된 URL과 서명된 쿠키 중 선택 (p. 124)
- 서명된 URL 사용 (p. 124)
- 서명된 쿠키 사용 (p. 141)
- Base64 인코딩 및 암호화를 위한 Linux 명령 및 OpenSSL 사용 (p. 155)
- 서명 URL에 대한 서명을 만드는 코드 예제 (p. 156)

프라이빗 콘텐츠 제공 개요

다음 그림에 표시된 것처럼, 프라이빗 콘텐츠에 대한 사용자 액세스를 두 가지 방법으로 제어할 수 있습니다.

1. CloudFront 엣지 캐시에 있는 파일에 대한 액세스 제한
2. 다음 중 하나를 수행하여 오리진의 파일에 대한 액세스를 제한합니다.
   - Amazon S3 버킷의 원본 액세스 ID(OAI) 설정
   - 프라이빗 HTTP 서버(사용자 지정 오리진)에 대한 사용자 지정 헤더 구성



CloudFront 엣지 캐시에 있는 파일에 대한 액세스 제한

사용자가 서명된 URL 또는 서명된 쿠키를 사용하여 파일에 액세스할 것을 요구하도록 CloudFront를 구성할 수 있습니다. 그런 다음 서명된 URL을 만들어 인증된 사용자에게 배포하거나 인증된 사용자용 최종 사용자에 `Set-Cookie` 헤더를 보내 서명된 쿠키를 설정하도록 하는 애플리케이션을 개발합니다. (사용자 몇 명에게 적은 수의 파일에 대한 장기 액세스 권한을 부여하려면 서명된 URL을 수동으로 만드는 방법도 있습니다.)

서명된 URL 또는 서명된 쿠키를 만들어 파일 액세스를 제어할 때 다음과 같은 제약 조건을 지정할 수 있습니다.

- URL의 효력이 사라지는 종료 날짜 및 시간
- (선택 사항) URL의 효력이 발생하는 날짜 및 시간
- (선택 사항) 콘텐츠에 액세스할 때 사용할 수 있는 컴퓨터의 IP 주소 또는 주소 범위

퍼블릭/프라이빗 키 페어의 프라이빗 키를 사용하여 서명된 URL 또는 서명된 쿠키의 일부분을 해시 및 서명합니다. 이 서명된 URL 또는 서명된 쿠키를 사용해 파일에 액세스하면 CloudFront은(는) URL 또는 쿠키의 서명된 부분과 서명되지 않은 부분을 비교합니다. 양쪽이 일치하지 않으면 CloudFront은(는) 파일을 제공하지 않습니다.

URL 또는 쿠키에 서명하려면 RSA-SHA1을 사용해야 합니다. CloudFront는 다른 알고리즘을 허용하지 않습니다.

Amazon S3 버킷에 있는 파일에 대한 액세스 제한

사용자가 CloudFront를 통해 액세스할 수는 있지만 Amazon S3 URL을 사용하여 직접 액세스할 수는 없도록 Amazon S3 버킷의 콘텐츠를 선택적으로 보호할 수 있습니다. 이렇게 하면 CloudFront를 우회하고 Amazon

S3 URL을 사용하여 액세스 제한 콘텐츠에 접근하는 사람을 막을 수 있습니다. 서명된 URL을 사용하기 위해 꼭 필요한 단계는 아니지만 이렇게 하는 것이 좋습니다.

사용자가 CloudFront URL을 통해 콘텐츠에 액세스하도록 요구하려면 다음 작업을 수행합니다.

- 원본 액세스 ID라는 특수 CloudFront 사용자를 만들어 CloudFront 배포와 연결합니다.
- 버킷의 파일을 읽을 수 있는 원본 액세스 ID 권한을 부여합니다.
- 나머지 모든 사람으로부터 Amazon S3 URL로 파일을 읽을 수 있는 권한을 제거합니다.

자세한 내용은 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오.

사용자 지정 오리진의 파일에 대한 액세스 제한

사용자 지정 오리진을 사용하는 경우 선택적으로 사용자 지정 헤더를 설정하여 액세스를 제한할 수 있습니다. CloudFront가 사용자 지정 오리진에서 파일을 가져오도록 하려면 파일이 공개적으로 액세스할 수 있어야 합니다. 그러나 사용자 지정 헤더를 사용하면 사용자가 직접 액세스할 수 없고 CloudFront를 통해서만 액세스할 수 있도록 콘텐츠에 대한 액세스를 제한할 수 있습니다. 서명된 URL을 사용하기 위해 꼭 필요한 단계는 아니지만 이렇게 하는 것이 좋습니다.

사용자가 CloudFront를 통해 콘텐츠에 액세스하도록 하려면 CloudFront 배포의 다음 설정을 변경합니다.

오리진 사용자 지정 헤더

오리진에 사용자 지정 헤더를 전달하도록 CloudFront를 구성합니다. 사용자 지정 헤더를 오리진 요청에 추가하도록 CloudFront 구성 (p. 249)을(를) 참조하십시오.

최종 사용자 프로토콜 정책

최종 사용자가 HTTPS를 사용하여 CloudFront에 액세스하도록 배포를 구성합니다. Viewer Protocol Policy (p. 45) 단원을 참조하십시오.

오리진 프로토콜 정책

CloudFront가 최종 사용자와 같은 프로토콜을 사용하여 오리진에 요청을 전달하도록 배포를 구성합니다. 오리진 프로토콜 정책 (p. 41) 단원을 참조하십시오.

변경을 수행한 후 이러한 헤더를 포함하는 요청만 수락하도록 사용자 지정 오리진의 애플리케이션을 업데이트합니다.

Viewer Protocol Policy(뷰어 프로토콜 정책)와 Origin Protocol Policy(오리진 프로토콜 정책)의 조합을 통해 사용자 정의 헤더가 최종 사용자와 오리진 사이에 암호화되도록 합니다. 하지만 주기적으로 다음을 수행하여 CloudFront가 오리진에 전달하는 사용자 지정 헤더를 교체하는 것이 좋습니다.

1. CloudFront를 업데이트하여 사용자 지정 오리진에 새 헤더 전달을 시작합니다.
2. 요청이 CloudFront에서 온다는 확인으로 애플리케이션을 업데이트하여 새 헤더를 허용합니다.
3. 최종 사용자 요청에 더 이상 교체하는 헤더가 포함되지 않는 경우, 요청이 CloudFront에서 온다는 확인으로 애플리케이션을 업데이트하여 새 헤더를 허용합니다.

작업 목록: 프라이빗 콘텐츠 제공

프라이빗 콘텐츠를 제공하도록 CloudFront를 구성하려면 다음 작업을 수행하십시오.

1. (권장 선택 사항) 사용자가 CloudFront를 통해서만 콘텐츠에 액세스할 수 있어야 합니다. 사용하는 방식은 Amazon S3 또는 사용자 지정 오리진 중 어떤 것을 사용하는지에 따라 다릅니다.

- Amazon S3 – 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176)단원을 참조하십시오.
- 사용자 지정 오리진 – 사용자 지정 오리진의 파일에 대한 액세스 제한 (p. 117) 단원을 참조하십시오.

사용자 지정 오리진은 Amazon EC2, 웹 사이트 엔드포인트로 구성된 Amazon S3 버킷, Elastic Load Balancing, 자체 HTTP 웹 서버를 포함합니다.

2. 서명된 URL 또는 서명된 쿠키를 생성할 때 사용할 AWS 계정을 지정합니다. 자세한 내용은 서명된 URL 및 서명된 쿠키(신뢰할 수 있는 서명자)를 만들 수 있는 AWS 계정 지정 (p. 118) 단원을 참조하십시오.
3. 서명된 쿠키를 설정하는 `Set-Cookie` 헤더 또는 서명된 URL을 보유한 권한 있는 사용자의 요청에 응답하는 애플리케이션을 작성합니다. 다음 주제 중 하나에 있는 단계를 따릅니다.

   - 서명된 URL 사용 (p. 124)
   - 서명된 쿠키 사용 (p. 141)

   어떤 방법을 사용할지 확실하지 않은 경우 서명된 URL과 서명된 쿠키 중 선택 (p. 124) 단원을 참조하십시오.

서명된 URL 및 서명된 쿠키(신뢰할 수 있는 서명자)를 만들 수 있는 AWS 계정 지정

주제

- 신뢰할 수 있는 서명자에 대해 CloudFront 키 페어 생성 (p. 119)
- CloudFront 프라이빗 키 재포맷(.NET 및 Java만 해당) (p. 121)
- 배포에 신뢰할 수 있는 서명자 추가 (p. 121)
- 신뢰할 수 있는 서명자의 활성 상태 확인(선택 사항) (p. 123)
- CloudFront 키 페어 교체 (p. 123)

서명된 URL 또는 서명된 쿠키를 만들려면 활성 CloudFront 키 페어가 있는 AWS 계정이 한 개 이상 필요합니다. 이 계정을 신뢰할 수 있는 서명자라고 합니다. 신뢰할 수 있는 서명자의 용도는 다음 두 가지입니다.

- 신뢰할 수 있는 서명자를 위한 AWS 계정 ID를 배포에 추가하는 즉시 CloudFront은(는) 파일에 액세스할 때 서명된 URL 또는 서명된 쿠키를 사용할 것을 사용자에게 요구하기 시작합니다.
- 서명된 URL 또는 서명된 쿠키를 만들 때는 신뢰할 수 있는 서명자의 키 페어에서 프라이빗 키를 사용하여 URL 또는 쿠키의 일부분에 서명합니다. 제한된 파일에 대한 요청이 들어오면 CloudFront은(는) URL 또는 쿠키의 서명된 부분을 서명되지 않은 부분과 비교하여 URL 또는 쿠키가 변조되지 않았음을 확인합니다. CloudFront은(는) URL 또는 쿠키가 유효한지, 의미가 있는지를 확인합니다(예: 만료 날짜 및 시간 이전인지 확인).

신뢰할 수 있는 서명자를 지정할 때 서명된 URL 또는 서명된 쿠키를 사용해야 하는 파일도 간접적으로 지정하게 됩니다.

- 웹 배포 – 신뢰할 수 있는 서명자를 캐시 동작에 추가합니다. 배포에 캐시 동작이 한 개뿐이라면 사용자는 반드시 서명된 URL 또는 서명된 쿠키를 사용하여 해당 배포와 연결된 파일에 액세스해야 합니다. 여러 가지 캐시 동작을 만들고 일부 캐시 동작에만 신뢰할 수 있는 서명자를 추가하는 경우, 일부 파일에 액세스할 때만 서명된 URL 또는 서명된 쿠키를 사용하도록 사용자에게 요구할 수 있습니다.
- RTMP 배포(서명된 URL만 해당) – 신뢰할 수 있는 서명자를 배포에 추가합니다. RTMP 배포에 신뢰할 수 있는 서명자를 추가한 뒤에는 사용자가 이 배포와 연결된 파일에 액세스할 때 서명된 URL을 사용해야 합니다.

> **Note**
>
> 배포에 대하여 신뢰할 수 있는 서명자를 지정하려면 CloudFront 콘솔 또는 CloudFront API 버전 2009-09-09 이상을 사용해야 합니다.

서명된 URL 또는 서명된 쿠키를 만들고 CloudFront 배포에 계정을 추가할 수 있는 계정을 지정하려면 다음 작업을 수행하십시오.

1. 신뢰할 수 있는 서명자로 사용할 AWS 계정을 결정합니다. 대부분의 CloudFront 고객은 배포를 만들 때 사용했던 계정을 사용합니다.
2. 1단계에서 선택한 계정별로 CloudFront 키 페어를 만듭니다. 자세한 내용은 신뢰할 수 있는 서명자에 대해 CloudFront 키 페어 생성 (p. 119) 단원을 참조하십시오.
3. .NET 또는 Java를 사용하여 서명된 URL 또는 서명된 쿠키를 만드는 경우, CloudFront 프라이빗 키를 다시 포맷해야 합니다. 자세한 내용은 CloudFront 프라이빗 키 재포맷(.NET 및 Java만 해당) (p. 121) 단원을 참조하십시오.
4. 서명된 URL 또는 서명된 쿠키를 만들려는 배포에서 신뢰할 수 있는 서명자의 AWS 계정 ID를 지정합니다. 자세한 내용은 배포에 신뢰할 수 있는 서명자 추가 (p. 121) 단원을 참조하십시오.
5. (선택 사항) 신뢰할 수 있는 서명자에게 활성 CloudFront 키 페어가 있음을 CloudFront가 인식하는지 확인합니다. 자세한 내용은 신뢰할 수 있는 서명자의 활성 상태 확인(선택 사항) (p. 123) 단원을 참조하십시오.

신뢰할 수 있는 서명자에 대해 CloudFront 키 페어 생성

CloudFront 서명된 URL 또는 서명된 쿠키를 만들 때 사용하는 각 AWS 계정(신뢰할 수 있는 서명자)에는 고유한 CloudFront 활성 키 페어가 있어야 합니다. Amazon EC2 키 페어를 CloudFront 키 페어로 대체할 수 없습니다. CloudFront 서명된 URL 또는 서명된 쿠키를 만들 때, 신뢰할 수 있는 서명자의 키 페어에 대한 키 페어 ID를 URL에 포함시킵니다. Amazon EC2에서는 키 페어 ID를 사용하도록 지정할 수 없습니다.

애플리케이션 보안을 위해 CloudFront 키 페어를 90일마다 또는 더 자주 변경하는 것이 좋습니다. 자세한 내용은 CloudFront 키 페어 교체 (p. 123) 단원을 참조하십시오.

키 페어를 만드는 방법은 다음과 같습니다.

- AWS Management 콘솔에서 키 페어를 만들고 프라이빗 키를 다운로드합니다. AWS Management 콘솔에서 CloudFront 키 페어를 만들려면 (p. 119) 절차를 참조하십시오.
- OpenSSL과 같은 애플리케이션을 사용하여 RSA 키 페어를 만들고 퍼블릭 키를 AWS Management 콘솔에 업로드합니다. AWS Management 콘솔에서 RSA 키 페어를 만들고 퍼블릭 키를 업로드하려면 (p. 120) 절차를 참조하십시오.

AWS Management 콘솔에서 CloudFront 키 페어를 만들려면

1. AWS 계정의 루트 자격 증명을 사용하여 AWS Management 콘솔에 로그인합니다.

   > **Important**
   >
   > IAM 사용자는 CloudFront 키 페어를 만들 수 없습니다. 키 페어를 만들려면 루트 자격 증명으로 로그인해야 합니다.
2. 계정 이름 메뉴에서 보안 자격 증명을 클릭합니다.
3. CloudFront 키 페어를 확장합니다.
4. 활성 키 페어가 하나만 있는지 확인합니다. 활성 키 페어가 이미 두 개라면 키 페어를 만들 수 없습니다.
5. 새 키 페어 생성을 클릭합니다.
6. 키 페어 만들기 대화 상자에서 프라이빗 키 파일 다운로드를 클릭합니다.
7. Opening<filename>(열기) 대화 상자에서 Save File(파일 저장)의 기본값을 사용하고, 확인을 클릭하여 CloudFront 키 페어의 프라이빗 키를 다운로드하고 저장합니다.

> **Important**
>
> CloudFront 키 페어의 프라이빗 키를 안전한 위치에 저장하고 파일에 대한 권한을 설정하여 원하는 관리자가 읽을 수 있도록 합니다. 다른 사람이 프라이빗 키를 가지게 되면 이들이 유효한 서명된 URL과 서명된 쿠키를 만들어 콘텐츠를 다운로드할 수 있게 됩니다. 프라이빗 키는 다시 가져올 수 없기 때문에 이를 잃어버리거나 삭제한 경우 새 CloudFront 키 페어를 만들어야 합니다.

8. 키 페어의 키 페어 ID를 기록합니다. (AWS Management 콘솔에서는 액세스 키라고 합니다.) 서명된 URL 또는 서명된 쿠키를 만들 때 이것을 사용합니다.

AWS Management 콘솔에서 RSA 키 페어를 만들고 퍼블릭 키를 업로드하려면

1. OpenSSL 또는 기타 도구를 사용해서 키 페어를 만듭니다.

   예를 들어 OpenSSL을 사용하는 경우, 다음 명령으로 길이가 4096비트인 키 페어를 만들고 `private_key.pem` 파일에 저장합니다.

   ```
   openssl genrsa -out private_key.pem 4096
   ```

   이렇게 만들어진 파일은 퍼블릭 키와 프라이빗 키를 모두 포함합니다. 그 파일에서 퍼블릭 키를 추출하려면 다음 명령을 실행합니다.

   ```
   openssl rsa -pubout -in private_key.pem -out public_key.pem
   ```

   퍼블릭 키(`public_key.pem`)는 이 절차의 뒷 부분에서 업로드하는 파일입니다.

   다음은 키에 대한 유의 사항입니다.

   - 키 페어는 SSH-2 RSA 키 페어여야 합니다.
   - 키 페어는 base64로 인코딩된 PEM 형식이어야 합니다.
   - 지원되는 키 길이는 1024, 2048, 4096비트입니다.
2. AWS 계정의 루트 자격 증명을 사용하여 AWS Management 콘솔에 로그인합니다.

   > **Important**
   >
   > IAM 사용자는 CloudFront 키 페어를 만들 수 없습니다. 키 페어를 만들려면 루트 자격 증명으로 로그인해야 합니다.
3. 계정 이름 메뉴에서 보안 자격 증명을 클릭합니다.
4. CloudFront 키 페어를 확장합니다.
5. 활성 키 페어가 하나만 있는지 확인합니다. 활성 키 페어가 이미 두 개라면 자신의 키 페어를 업로드할 수 없습니다.
6. Upload Your Own Key Pair(자체 키 페어 업로드)를 클릭합니다.
7. Upload Your Own Key Pair(자체 키 페어 업로드) 대화 상자에서 Choose File(파일 선택)을 클릭하고 1 단계에서 만든 퍼블릭 키 파일을 선택합니다.
8. 업로드를 클릭합니다.

   Upload Key Pair(키 페어 업로드) 대화 상자는 사라지고 새로운 키 페어가 CloudFront 키 페어 목록의 상단에 나타납니다.
9. 키 페어의 키 페어 ID를 기록합니다. (AWS Management 콘솔에서는 액세스 키라고 합니다.) 서명된 URL 또는 서명된 쿠키를 만들 때 이것을 사용합니다.

CloudFront 프라이빗 키 재포맷(.NET 및 Java만 해당)

.NET 또는 Java를 사용하여 서명된 URL 또는 서명된 쿠키를 만드는 경우, 기본 .pem 형식의 키 페어에서 프라이빗 키를 사용하여 서명을 만들 수 없습니다.

- .NET 프레임워크 – 프라이빗 키를 .NET 프레임워크에서 사용하는 XML 형식으로 변환합니다. 몇 가지 도구를 사용할 수 있습니다.
- Java – 프라이빗 키를 DER 형식으로 변환합니다. 이렇게 하려면 OpenSSL을 사용하면 됩니다.

  ```
  $ openssl pkcs8 -topk8 -nocrypt -in origin.pem -inform PEM -out new.der -
  outform DER
  ```

  인코더가 정확하게 작동하려면 Bouncy Castle Java 암호화 API에 대한 jar를 프로젝트에 추가한 다음 Bouncy Castle 공급자를 추가합니다.

배포에 신뢰할 수 있는 서명자 추가

신뢰할 수 있는 서명자란 배포에 대해 서명된 URL 및 서명된 쿠키를 만들 수 있는 AWS 계정입니다. 기본적으로, 해당 배포를 만든 계정을 포함하여 어떤 계정도 서명된 URL 또는 서명된 쿠키를 만들 수 없습니다. 신뢰할 수 있는 서명자로 사용할 AWS 계정을 지정하려면 배포에 해당 계정을 추가하십시오.

- 웹 배포 – 신뢰할 수 있는 서명자에는 캐시 동작이 연결되어 있습니다. 이렇게 하면 같은 배포의 일부 파일에 대해서만 서명된 URL 또는 서명된 쿠키를 요구할 수 있습니다. 신뢰할 수 있는 서명자는 해당 캐시 동작과 연결된 파일에 대해서만 서명된 URL 또는 쿠키를 만들 수 있습니다. 예를 들어, 캐시 동작 하나당 신뢰할 수 있는 서명자가 하나씩 있고 다른 캐시 동작에는 다른 신뢰할 수 있는 서명자가 있다면 양쪽의 신뢰할 수 있는 서명자 모두 상대편의 캐시 동작과 연결된 파일에 대해 서명된 URL 또는 쿠키를 만들 수 없습니다.
- RTMP 배포(서명된 URL만 해당) – 신뢰할 수 있는 서명자는 배포와 연결되어 있습니다. RTMP 배포에 신뢰할 수 있는 서명자를 추가한 뒤에는 사용자가 이 배포와 연결된 객체에 액세스할 때 서명된 URL 또는 서명된 쿠키를 사용해야 합니다.

  > **Important**
  >
  > 경로 패턴 및 시퀀스를 신중하게 정의하여 콘텐츠 액세스 권한을 실수로 사용자에게 부여하거나 일부 공개 콘텐츠에 사용자가 액세스하는 일이 없도록 해야 합니다. 예를 들어, 하나의 요청이 두 캐시 동작의 경로 패턴과 일치한다고 가정합니다. 첫 번째 캐시 동작에는 서명된 URL 또는 서명된 쿠키가 필요하지 않고, 두 번째 캐시 동작에는 필요합니다. CloudFront에서는 첫 번째와 연결된 캐시 동작을 처리하므로 사용자는 서명된 URL 또는 서명된 쿠키 없이도 파일에 액세스할 수 있습니다.

경로 패턴에 대한 자세한 내용은 경로 패턴 (p. 44)을 참조하십시오.

> **Important**
>
> 콘텐츠 배포를 위해 이미 사용 중인 배포를 업데이트하려는 경우, 파일에 대해 서명된 URL 또는 서명된 쿠키를 만들 준비가 되었을 때만 신뢰할 수 있는 서명자를 추가하십시오. 그렇지 않은 경우 CloudFront은(는) 요청을 거절합니다.
>
> - 웹 배포 – 웹 배포에 대한 캐시 동작에 신뢰할 수 있는 서명자를 추가한 뒤에는 사용자가 이 캐시 동작과 연결된 파일에 액세스할 때 서명된 URL 또는 서명된 쿠키를 사용해야 합니다.
> - RTMP 배포(서명된 URL만 해당) – RTMP 배포에 신뢰할 수 있는 서명자를 추가한 뒤에는 사용자가 이 배포와 연결된 파일에 액세스할 때 서명된 URL을 사용해야 합니다.

신뢰할 수 있는 서명자의 최대 수는 배포의 유형에 따라 달라집니다.

- 웹 배포 – 캐시 동작당 최대 5개
- RTMP 배포 – 배포당 최대 5개

CloudFront 콘솔 또는 CloudFront API를 사용하여 신뢰할 수 있는 서명자를 배포에 추가할 수 있습니다. 다음 주제를 참조하십시오.

- CloudFront 콘솔을 사용하여 배포에 신뢰할 수 있는 서명자 추가 (p. 122)
- CloudFront API를 사용하여 배포에 신뢰할 수 있는 서명자 추가 (p. 122)

CloudFront 콘솔을 사용하여 배포에 신뢰할 수 있는 서명자 추가

CloudFront 콘솔을 사용하여 신뢰할 수 있는 서명자를 배포에 추가하려면

1. 해당 배포를 만든 AWS 계정만 신뢰할 수 있는 서명자로 사용하려는 경우, 2단계로 건너뜁니다.

   다른 AWS 계정을 사용하려는 경우, 계정별로 AWS 계정 ID를 가져옵니다.

   a. https://console.aws.amazon.com/console/home에서 신뢰할 수 있는 서명자로 사용할 계정으로 AWS Management 콘솔에 로그인합니다.
   b. 콘솔의 오른쪽 상단 모서리에서 계정과 연결된 이름을 클릭하고 내 계정을 클릭합니다.
   c. 계정 설정 아래의 계정 ID를 기록해 둡니다.
   d. AWS Management 콘솔에서 로그아웃합니다.
   e. 신뢰할 수 있는 서명자로 사용하려는 나머지 계정에 대해 a단계부터 d단계까지 반복합니다.
2. https://console.aws.amazon.com/cloudfront/에서 Amazon CloudFront 콘솔을 열고, 신뢰할 수 있는 서명자를 추가하려는 배포를 만들 때 사용했던 계정으로 로그인합니다.
3. 배포 ID를 클릭합니다.
4. 편집 모드로 변경:

   - 웹 배포 – 동작 탭을 클릭하고 편집할 동작을 클릭한 다음 편집을 클릭합니다.
   - RTMP 배포 – 편집을 클릭합니다.
5. Restrict Viewer Access (Use Signed URLs or Signed Cookies)(최종 사용자 액세스 제한(서명된 URL 또는 서명된 쿠키 사용))에서 예를 클릭합니다.
6. 신뢰할 수 있는 서명자에서 시나리오에 대한 확인란을 선택합니다.

   - 셀프 – 현재 계정(배포를 만들 때 사용한 계정)을 사용하려는 경우 이 확인란을 선택합니다.
   - Specify Accounts(계정 지정) – 다른 AWS 계정을 사용하려는 경우 이 확인란을 선택합니다.
7. Specify Accounts(계정 지정) 확인란을 선택한 경우 AWS 계정 번호 필드에 AWS 계정 ID를 입력합니다. 이것은 이 절차의 첫 번째 단계에서 얻은 계정 ID입니다. 계정 ID를 한 줄에 하나씩 입력합니다.
8. 예, 편집합니다를 클릭합니다.
9. 웹 배포에 신뢰할 수 있는 서명자를 추가할 때 캐시 동작이 둘 이상인 경우 4단계부터 8단계까지 반복합니다.

CloudFront API를 사용하여 배포에 신뢰할 수 있는 서명자 추가

CloudFront API를 사용하여 신뢰할 수 있는 서명자에 대한 AWS 계정 ID를 기존 배포에 추가하거나, 신뢰할 수 있는 서명자가 포함된 배포를 새로 만들 수 있습니다. 두 경우 모두 `TrustedSigners` 요소에 값을 지정합니다. 웹 배포의 경우, `TrustedSigners` 요소를 하나 이상의 캐시 동작에 추가합니다. RTMP 배포의 경우, 배포에 `TrustedSigners` 요소를 추가합니다.

Amazon CloudFront API Reference에서 다음 주제를 참조하십시오.

- 새로운 웹 배포 생성 – CreateDistribution
- 기존 웹 배포 업데이트 – UpdateDistribution
- 새로운 RTMP 배포 만들기 – CreateStreamingDistribution
- 기존 RTMP 배포 업데이트 – UpdateStreamingDistribution

신뢰할 수 있는 서명자의 활성 상태 확인(선택 사항)

신뢰할 수 있는 서명자를 배포에 추가한 뒤, 서명자가 활성화되었는지 확인할 수 있습니다. 신뢰할 수 있는 서명자가 활성화되려면 다음 조건이 참이어야 합니다.

- AWS 계정에 활성 키 페어가 하나 이상 있어야 합니다. 키 페어를 돌아가며 사용하는 경우, 계정은 일시적으로 기존 키 페어와 새 키 페어 두 가지를 가지게 됩니다.
- CloudFront가 활성 키 페어를 인식하고 있어야 합니다. 키 페어를 만든 후 CloudFront가 키 페어의 존재를 인식하기까지 시간이 약간 걸립니다.

> **Note**
>
> 현재는 배포에 대해 신뢰할 수 있는 활성 서명자 목록을 보려면 CloudFront API를 사용해야 합니다. CloudFront 콘솔에서는 신뢰할 수 있는 활성 서명자 목록을 볼 수 없습니다.

CloudFront API를 사용하여 신뢰할 수 있는 활성 서명자 확인

신뢰할 수 있는 서명자 중 활성 키 페어가 있는 서명자(신뢰할 수 있는 활성 서명자)를 파악하려면 해당 배포에서 `ActiveTrustedSigners` 요소의 값을 검토합니다. 이 요소는 해당 배포가 신뢰할 수 있는 서명자로 인식하는 각 계정의 AWS 계정 ID를 나열합니다. 신뢰할 수 있는 서명자에 활성 CloudFront 키 페어가 한 개 이상 있는 경우, `ActiveTrustedSigners` 요소는 키 페어 ID도 나열합니다. 자세한 내용은 Amazon CloudFront API Reference에서 다음 주제를 참조하십시오.

- 웹 배포 – GetDistribution
- RTMP 배포 – GetStreamingDistribution

CloudFront 키 페어 교체

AWS는 활성 CloudFront 키 페어를 90일마다 교체(변경)할 것을 권장합니다. 아직 만료되지 않은 서명된 URL 또는 서명된 쿠키를 무효화하지 않고 URL 또는 쿠키를 만들기 위해 사용 중인 CloudFront 키 페어를 교체하려면 다음 작업을 수행하십시오.

1. 서명된 URL을 만들 때 사용하는 계정별로 새로운 키 페어를 만듭니다. 자세한 내용은 신뢰할 수 있는 서명자에 대해 CloudFront 키 페어 생성 (p. 119) 단원을 참조하십시오.
2. CloudFront가 새로운 키 페어를 인식하는지 확인합니다. 자세한 내용은 신뢰할 수 있는 서명자의 활성 상태 확인(선택 사항) (p. 123) 단원을 참조하십시오.
3. 애플리케이션을 업데이트하고 새로운 키 페어의 프라이빗 키를 사용하여 서명을 만듭니다.
4. 새로운 프라이빗 키로 서명한 URL 또는 쿠키가 작동하는지 확인합니다.
5. 기존 CloudFront 키 페어로 서명한 URL 또는 쿠키의 만료 날짜가 지날 때까지 기다립니다.
6. 기존 CloudFront 키 페어를 비활성으로 변경합니다.
   a. 키 페어를 비활성화할 AWS 계정의 루트 자격 증명을 사용하여 AWS Management 콘솔에 로그인합니다.
   b. 계정 이름 메뉴에서 보안 자격 증명을 클릭합니다.
   c. CloudFront 키 페어를 확장합니다.
   d. 특정 키 페어를 선택한 후 Make Inactive(비활성화)를 선택합니다.
   e. 키 페어를 비활성화할 AWS 계정별로 a단계부터 d단계까지 반복합니다.
7. 새로운 프라이빗 키로 서명한 URL 또는 쿠키가 작동하는지 다시 확인합니다.
8. 기존 CloudFront 키 페어를 삭제합니다.
   a. 보안 자격 증명 페이지를 참조하십시오.
   b. CloudFront 키 페어를 확장합니다.
   c. 특정 키 페어를 선택한 후 삭제를 선택합니다.

9. 저장된 기존 프라이빗 키를 삭제합니다.

서명된 URL과 서명된 쿠키 중 선택

CloudFront 서명된 URL 및 서명된 쿠키는 기본 기능이 같습니다. 바로 콘텐츠에 액세스할 수 있는 대상을 제어하는 기능입니다. CloudFront를 통해 프라이빗 콘텐츠를 제공 중이며 서명된 URL 또는 서명된 쿠키 중 하나를 결정해야 하는 경우, 다음 사항을 고려하십시오.

다음과 같은 경우 서명된 URL을 사용합니다.

- RTMP 배포를 사용하려는 경우. RTMP 배포에는 서명된 쿠키를 사용할 수 없습니다.
- 애플리케이션 설치 파일을 다운로드하는 등 개별 파일에 대한 액세스를 제한하고 싶은 경우
- 사용자가 쿠키를 지원하지 않는 클라이언트(예: 사용자 지정 HTTP 클라이언트)를 사용하는 경우

다음과 같은 경우 서명된 쿠키를 사용합니다.

- HLS 형식의 비디오 파일 전체 또는 웹 사이트의 구독자 영역에 있는 파일 전체 등 제한된 파일 여러 개에 대한 액세스 권한을 제공하려는 경우
- 현재의 URL을 변경하고 싶지 않은 경우

현재 서명된 URL을 사용하지 않는데 URL에 다음과 같은 쿼리 문자열 파라미터가 들어 있다면 서명된 URL 또는 서명된 쿠키를 사용할 수 없습니다.

- `Expires`
- `Policy`
- `Signature`
- `Key-Pair-Id`

CloudFront는 이러한 쿼리 문자열 파라미터가 포함된 URL을 서명된 URL로 간주하기 때문에 서명된 쿠키를 살펴보지 않을 것입니다.

서명된 URL과 서명된 쿠키 모두 사용

같은 파일에 대한 액세스를 제어할 때 서명된 URL과 서명된 쿠키를 모두 사용하고 최종 사용자는 서명된 URL로 파일을 요청하는 경우, CloudFront는 서명된 URL만을 기준으로 최종 사용자에게 파일을 반환할지 여부를 판단합니다.

서명된 URL 사용

주제

- 서명된 URL에 대해 미리 준비된(canned) 정책과 사용자 지정 정책 중 선택 (p. 125)
- 서명된 URL의 작동 방식 (p. 125)
- 서명된 URL의 유효 기간 선택 (p. 126)
- CloudFront가 서명된 URL에서 만료 날짜 및 시간을 확인하는 경우 (p. 126)
- 샘플 코드 및 타사 도구 (p. 127)
- 미리 준비된 정책을 사용하여 서명된 URL 생성 (p. 127)
- 사용자 지정 정책을 사용한 서명된 URL 생성 (p. 133)

서명된 URL에는 만료 날짜 및 시간 같은 추가 정보가 포함되므로 콘텐츠에 대한 액세스를 보다 세부적으로 제어할 수 있습니다. 이러한 추가 정보는 미리 준비된(canned) 정책 또는 사용자 지정 정책에 따라 정책 설명

에 나타납니다. 미리 준비된(canned) 정책과 사용자 지정 정책 간의 차이점은 이어지는 두 단원에 설명되어 있습니다.

> **Note**
>
> 같은 배포에 대해 미리 준비된(canned) 정책과 사용자 지정 정책으로 각각 서명된 URL을 만들 수 있습니다.

서명된 URL에 대해 미리 준비된(canned) 정책과 사용자 지정 정책 중 선택

서명된 URL을 만들 때 JSON 형식의 정책 설명을 작성하여 서명된 URL의 제약 조건(예: URL의 유효 기간)을 지정합니다. 미리 준비된(canned) 정책 또는 사용자 지정 정책을 사용할 수 있습니다. 다음은 미리 준비된(canned) 정책과 사용자 지정 정책을 비교한 것입니다.

설명	미리 준비된(canned) 정책	사용자 지정 정책
정책 설명은 여러 파일에 재사용할 수 있습니다. 정책 설명을 재사용하려면 `Resource` 객체에 와일드카드 문자를 사용해야 합니다. 자세한 내용은 사용자 지정 정책을 사용하는 서명된 URL에 대한 정책 설명에서 지정한 값 (p. 137) 단원을 참조하십시오.	아니요	예
날짜 및 시간을 지정하여 사용자가 콘텐츠에 액세스를 시작할 수 있습니다.	아니요	예(선택 사항)
사용자의 콘텐츠 액세스를 중단할 날짜 및 시간을 지정할 수 있습니다.	예	예
콘텐츠에 액세스할 수 있는 사용자의 IP 주소 또는 IP 주소 범위를 지정할 수 있습니다.	아니요	예(선택 사항)
서명된 URL에는 base64 인코딩 버전의 정책이 포함되므로 URL 길이가 더 깁니다.	아니요	예

미리 준비된(canned) 정책으로 서명된 URL을 만드는 방법에 대한 자세한 내용은 미리 준비된 정책을 사용하여 서명된 URL 생성 (p. 127) 단원을 참조하십시오.

사용자 지정 정책으로 서명된 URL을 만드는 방법에 대한 자세한 내용은 사용자 지정 정책을 사용한 서명된 URL 생성 (p. 133) 단원을 참조하십시오.

서명된 URL의 작동 방식

여기서는 서명된 URL에 알맞게 CloudFront 및 Amazon S3을(를) 구성하는 방법과 사용자가 서명된 URL로 파일을 요청할 때 CloudFront이(가) 대응하는 방식을 살펴봅니다.

1. CloudFront 배포에서 신뢰할 수 있는 서명자(서명된 URL을 만들 권한이 있는 AWS 계정)를 하나 이상 지정합니다.

   자세한 내용은 서명된 URL 및 서명된 쿠키(신뢰할 수 있는 서명자)를 만들 수 있는 AWS 계정 지정 (p. 118) 단원을 참조하십시오.
2. 애플리케이션을 개발하여 사용자에게 콘텐츠 액세스를 허용할지 여부를 판단하고, 액세스를 제한할 애플리케이션 부분 또는 파일에 대해 서명된 URL을 만듭니다. 자세한 내용은 다음 항목을 참조하십시오.

   - 미리 준비된 정책을 사용하여 서명된 URL 생성 (p. 127)

- 사용자 지정 정책을 사용한 서명된 URL 생성 (p. 133)

3. 사용자가 서명된 URL이 필요한 파일을 요청합니다.
4. 애플리케이션은 해당 사용자에게 파일 액세스 권한이 있는지 확인합니다. 사용자가 로그인했는지, 콘텐츠 액세스 비용을 지불했는지, 기타 액세스 요구 사항을 충족했는지 등을 확인합니다.
5. 애플리케이션은 서명된 URL을 만들어 사용자에게 반환합니다.
6. 사용자는 서명된 URL을 통해 콘텐츠를 다운로드하거나 스트리밍할 수 있습니다.

   이 단계는 자동으로 진행되기 때문에 사용자는 콘텐츠에 액세스하기 위해 아무것도 하지 않아도 됩니다. 예를 들어, 사용자가 웹 브라우저에서 콘텐츠에 액세스한다면 애플리케이션은 서명된 URL을 브라우저에 반환합니다. 브라우저는 즉시 서명된 URL을 사용하여 사용자 개입 없이 CloudFront 엣지 캐시의 파일에 액세스합니다.
7. CloudFront는 퍼블릭 키를 사용하여 서명을 확인하고 URL이 아직 변조되지 않았음을 확인합니다. 유효하지 않은 서명인 경우 요청을 거부합니다.

   서명이 유효한 경우, CloudFront는 URL의 정책 설명을 보고 아직 유효한 정책임을 확인합니다. 미리 준비된(canned) 정책을 사용하는 경우에는 정책 설명을 구성합니다. 예를 들어, URL의 시작 및 종료 날짜와 시간을 지정했다면 CloudFront는 액세스가 허용되는 시간 동안 콘텐츠 액세스를 시도한 사용자를 확인합니다.

   요청이 정책 설명의 요구 사항을 충족하는 경우, CloudFront는 스탠다드 작업을 수행합니다. 다시 말해 파일이 이미 엣지 캐시에 있는지 여부를 판단하고, 필요에 따라 요청을 오리진으로 전달하고, 사용자에게 파일을 반환합니다.

   Note

   서명된 CloudFront URL에는 추가 쿼리 문자열 인수를 포함할 수 없습니다. 서명된 URL을 생성한 후 생성한 URL에 쿼리 문자열을 추가하면 해당 URL이 HTTP 403 상태를 반환합니다.

서명된 URL의 유효 기간 선택

단시간(최소 몇 분) 동안만 유효한 서명된 URL을 사용하여 프라이빗 콘텐츠를 배포할 수 있습니다. 이렇게 단시간 동안 유효한 서명된 URL은 영화 대여 또는 음악 다운로드 등을 주문 방식으로 고객에게 제공하는 등 특정 목적으로 즉석에서 콘텐츠를 배포할 때 유용합니다. 서명된 URL의 효력을 단시간 동안만 유지하려는 경우, 이를 자동으로 만드는 애플리케이션을 개발할 수 있습니다. 사용자가 파일을 다운로드하거나 미디어 파일을 재생하기 시작하면 CloudFront는 URL의 만료 시간을 현재 시간과 비교하여 URL이 아직 유효한지 파악합니다.

또한 장시간(몇 년 정도) 동안 유효한 서명된 URL을 사용하여 프라이빗 콘텐츠를 배포할 수도 있습니다. 장시간 동안 유효한 서명된 URL은 알려진 사용자에게 프라이빗 콘텐츠를 배포할 때 유용합니다. 투자자에게 사업 계획을 배포하거나 직원에게 교육 자료를 배포하는 경우가 여기에 해당합니다. 이러한 장기간 서명된 URL을 만드는 애플리케이션을 개발하거나 비공개 콘텐츠 구성을 위한 도구 및 코드 예제 (p. 454)에 나열된 타사 GUI 도구를 사용할 수 있습니다.

CloudFront가 서명된 URL에서 만료 날짜 및 시간을 확인하는 경우

CloudFront는 서명된 URL의 만료 날짜와 시간을 확인할 때 해당 URL이 웹 배포용인지 아니면 RTMP 배포용인지에 따라 URL이 아직 유효한지 여부를 파악할 수 있습니다.

- 웹 배포 – CloudFront는 HTTP 요청이 있을 때 서명된 URL의 만료 날짜 및 시간을 확인합니다. 클라이언트가 만료 시간 직전에 대용량 파일을 다운로드하기 시작한 경우, 다운로드 도중 만료 시간이 지나도 다운로드는 완료됩니다. TCP 연결이 끊어진 경우, 클라이언트가 만료 시간 이후에 다운로드를 다시 시작하는 것은 불가능합니다.

  클라이언트가 범위 GET을 사용하여 파일을 작은 조각으로 가져오는 경우, 만료 시간 이후의 GET 요청은 실패합니다. 범위 GET에 대한 자세한 내용은 CloudFront에서 객체에 대한 부분적인 요청을 처리하는 방법 (Range GET) (p. 214)을 참조하십시오.

- RTMP 배포 – CloudFront는 재생 이벤트가 시작될 때 서명된 URL의 만료 시간을 확인합니다. 만료 시간이 지나기 전에 클라이언트가 미디어 파일을 재생하기 시작하는 경우, CloudFront는 전체 미디어 파일을 재생합니다. 그러나 미디어 플레이어에 따라서는 중지했다가 재시작하면 다른 재생 이벤트가 트리거될 수 있습니다. 미디어 파일의 다른 위치로 건너뛰면 다른 재생 이벤트가 트리거됩니다. 만료 시간이 지난 후에 후속 재생 이벤트가 발생하는 경우, CloudFront는 미디어 파일을 제공하지 않습니다.

  Note

  Adobe는 2020년 12월 31일까지 Flash 지원을 종료하기로 결정했습니다. 결과적으로 Amazon CloudFront는 더 이상 Adobe Flash Media Server를 지원하지 않으며 2020년 12월 31일까지 RTMP(Real-Time Messaging Protocol)를 더 이상 배포하지 않을 것입니다. 자세한 내용은 Amazon CloudFront 토론 포럼의 발표 전문을 읽어 보십시오.

샘플 코드 및 타사 도구

서명된 URL의 해시 및 서명된 부분을 만드는 샘플 코드에 대해서는 다음 주제를 참조하십시오.

- Perl을 사용한 URL 서명 생성 (p. 156)
- PHP를 사용한 URL 서명 생성 (p. 164)
- C# 및 .NET Framework를 사용한 URL 서명 생성 (p. 169)
- Java를 사용한 URL 서명 생성 (p. 174)

서명된 URL 생성 등 프라이빗 콘텐츠를 지원하는 타사 도구에 대한 자세한 내용은 비공개 콘텐츠 구성을 위한 도구 및 코드 예제 (p. 454)을 참조하십시오.

미리 준비된 정책을 사용하여 서명된 URL 생성

미리 준비된 정책을 사용하여 서명된 URL을 만들려면 다음 절차를 수행하십시오.

미리 준비된 정책을 사용하여 서명된 URL을 만들려면

1. .NET 또는 Java를 사용하여 서명된 URL을 만드는 중인데 키 페어의 프라이빗 키를 기본 .pem 형식에서 .NET 또는 Java와 상환되는 형식으로 다시 포맷하지 않았다면 지금 포맷하세요. 자세한 내용은 CloudFront 프라이빗 키 재포맷(.NET 및 Java만 해당) (p. 121) 단원을 참조하십시오.
2. 지정된 순서대로 다음 값을 연결하고 사이의 공백(탭과 줄바꿈 문자 포함)을 제거합니다. 애플리케이션 코드의 문자열에 이스케이프 문자를 포함해야 할 수도 있습니다. 모든 값에는 일종의 문자열이 있습니다. 다음 두 가지 예처럼 각 부분을 숫자(1)로 입력합니다.

   1 ***### ## URL***

   기본 URL이란 서명된 URL을 사용하지 않고 파일에 액세스할 때 사용할 CloudFront URL을 말합니다. 자체 쿼리 문자열 파라미터가 있는 경우 여기에 포함됩니다. 웹 배포의 URL 형식에 대한 자세한 내용은 CloudFront에서 파일에 대한 URL 형식 사용자 지정 (p. 79)을 참조하십시오.

   다음 예는 웹 배포에 대해 지정하는 값을 보여줍니다.

   - 다음 CloudFront URL은 웹 배포의 파일을 위한 것입니다(CloudFront 도메인 이름 사용). `image.jpg`는 `images` 디렉터리에 있습니다. URL 파일의 경로는 HTTP 서버 또는 Amazon S3 버킷의 파일 경로와 일치해야 합니다.

     ```
     http://d111111abcdef8.cloudfront.net/images/image.jpg
     ```
   - 다음 CloudFront URL은 쿼리 문자열을 포함합니다.

     ```
     http://d111111abcdef8.cloudfront.net/images/image.jpg?size=large
     ```
   - 다음 CloudFront URL은 웹 배포의 파일을 위한 것입니다. 둘 다 대체 도메인 이름을 사용하며 두 번째는 쿼리 문자열을 포함합니다.

```
http://www.example.com/images/image.jpg

http://www.example.com/images/image.jpg?color=red
```

- 다음 CloudFront URL은 대체 도메인 이름과 HTTPS 프로토콜을 사용하는 웹 배포의 파일을 위한 것입니다.

```
https://www.example.com/images/image.jpg
```

RTMP 배포의 경우, 다음은 MP4 및 FLV라는 두 가지 비디오 형식의 파일에 대한 예제입니다.

- MP4 – `mp4:sydney-vacation.mp4`
- FLV – `sydney-vacation`
- FLV – `sydney-vacation.flv`

> **Note**
>
> .flv 파일의 경우 플레이어에 따라 `.flv` 파일 이름 확장자의 포함 여부가 결정됩니다. MP3 오디오 파일 또는 H.264/MPEG-4 비디오 파일을 제공하려면 파일 이름에 `mp3:` 또는 `mp4:` 접두사를 적용해야 할 수 있습니다. 일부 미디어 플레이어에서는 자동으로 해당 접두사가 추가되도록 구성되어 있습니다. 그러한 미디어 플레이어에서는 또한 파일 확장명 없이(예: sydney-vacation.mp4 대신 sydney-vacation) 파일 이름을 지정하도록 요구할 수 있습니다.

2 ?

The ?는 기본 URL 뒤에 쿼리 문자열 파라미터가 있음을 나타냅니다. 자체 쿼리 문자열 파라미터가 없는 경우에는 ?를 포함하십시오.

3 ***## ## ### ####(## ##)***&

이 값은 선택 사항입니다. 예를 들어, 자체 쿼리 문자열 파라미터로

```
color=red&size=medium
```

을 추가하고 싶은 경우 ? 뒤(**2** 참조)와 `Expires` 파라미터 앞 사이에 추가합니다. 드물지만 `Key-Pair-Id` 뒤에 쿼리 문자열 파라미터를 넣어야 하는 경우도 있습니다.

> **Important**
>
> 파라미터는 `Expires`, `Signature` 또는 `Key-Pair-Id`라는 이름으로 지정할 수 없습니다.

자체 파라미터를 추가하는 경우, 마지막 파라미터를 포함하여 각 파라미터 뒤에 &를 추가합니다.

4 **Expires=*Unix ## ##(#) # ## ###(UTC) ### ## # ##***

URL에서 파일에 대한 액세스 허용을 중지하게 할 날짜 및 시간입니다.

Unix 시간 형식(초) 및 협정 세계시(UTC) 기준의 URL 만료 날짜 및 시간을 지정합니다. 예를 들면 2013년 1월 1일 오전 10시(UTC)를 Unix 시간 형식의 1357034400으로 변환합니다. 에포크(epoch) 시간을 사용하려면 날짜에 대해 32비트 정수를 사용합니다. 이 정수를 2147483647(2038년 1월 19일 03:14:07 UTC)를 넘을 수 없습니다. UTC에 대한 자세한 내용은 RFC 3339, 인터넷의 날짜 및 시간: 타임스탬프, http://tools.ietf.org/html/rfc3339를 참조하십시오.

5 **&Signature=*## ### ## # ### ##***

JSON 정책 설명을 해시, 서명 및 base64로 인코딩한 버전입니다. 자세한 내용은 미리 준비된 정책을 사용하는 서명된 URL에 대한 서명 생성 (p. 129) 단원을 참조하십시오.

❻ **&Key-Pair-Id=*### ### # #### # ### ## CloudFront # ## Id***

APKA9ONS7QCOWEXAMPLE와 같은 활성 CloudFront 키 페어의 ID. CloudFront는 서명된 URL을 확인할 때 사용할 퍼블릭 키를 CloudFront 키 페어 ID로 판단합니다. CloudFront는 서명의 정보를 정책 설명의 정보와 비교하고 URL이 변조되지 않았음을 확인합니다.

CloudFront 서명된 URL에 포함한 키 페어 ID는 신뢰할 수 있는 서명자 중 하나의 활성 키 페어 ID여야 합니다.

- 웹 배포 – 키 페어는 캐시 동작의 신뢰할 수 있는 서명자 중 하나인 AWS 계정과 연결되어야 합니다.
- RTMP 배포 – 키 페어는 배포에 대한 신뢰할 수 있는 서명자 중 하나인 AWS 계정과 연결되어야 합니다.

자세한 내용은 서명된 URL 및 서명된 쿠키(신뢰할 수 있는 서명자)를 만들 수 있는 AWS 계정 지정 (p. 118) 단원을 참조하십시오.

CloudFront 키 페어를 교체하면서 키 페어 하나를 비활성화하고 프로그래밍 방식으로 서명된 URL을 만들려는 경우, 신뢰할 수 있는 서명자 중 하나의 새로운 활성 키 페어를 사용하도록 애플리케이션을 업데이트해야 합니다. 서명된 URL을 수동으로 만들 때는 새로 만들어야 합니다. 키 페어 교체에 대한 자세한 내용은 CloudFront 키 페어 교체 (p. 123)를 참조하십시오.

웹 배포의 서명된 URL 예제:

❶ **http://d111111abcdef8.cloudfront.net/image.jpg** ❷
? ❸ **color=red&size=medium&** ❹ **Expires=1357034400**
❺ **&Signature=nitfHRCrtziwO2HwPfWw~yYDhUF5EwRunQA-j19DzZrvDh6hQ73lDx~~ar3UocvvRQVw6EkC~GdpGQyyOSKQim-TxAnW7d8F5Kkai9HVx0FIu-5jcQb0UEmatEXAMPLE3ReXySpLSMj0yCd3ZAB4UcBCAqEijkytL6f3fVYNGQI6**
❻ **&Key-Pair-Id=APKA9ONS7QCOWEXAMPLE**

RTMP 배포의 서명된 URL 예제:

❶ **videos/mediafile.flv** ❷ **?** ❸ **color=red&size=medium&** ❹
Expires=1357034400 ❺ **&Signature=nitfHRCrtziwO2HwPfWw~yYDhUF5EwRunQA-j19DzZrvDh6hQ73lDx~~ar3UocvvRQVw6EkC~GdpGQyyOSKQim-TxAnW7d8F5Kkai9HVx0FIu-5jcQb0UEmatEXAMPLE3ReXySpLSMj0yCd3ZAB4UcBCAqEijkytL6f3fVYNGQI6**
❻ **&Key-Pair-Id=APKA9ONS7QCOWEXAMPLE**

미리 준비된 정책을 사용하는 서명된 URL에 대한 서명 생성

미리 준비된 정책을 사용하는 서명된 URL의 서명을 만들려면 다음 절차를 수행하십시오.

1. 정책 설명을 만듭니다. 미리 준비된 정책을 사용하는 서명된 URL의 정책 설명 생성 (p. 129) 단원을 참조하십시오.
2. 정책 설명에 서명하여 서명을 만듭니다. 미리 준비된 정책을 사용하는 서명된 URL에 대한 서명 생성 (p. 131) 단원을 참조하십시오.

미리 준비된 정책을 사용하는 서명된 URL의 정책 설명 생성

미리 준비된 정책을 사용하여 서명된 URL을 만드는 경우, `Signature` 파라미터는 정책 설명의 해시 및 서명된 버전입니다. 미리 준비된 정책을 사용하는 서명된 URL의 경우, 사용자 지정 정책을 사용하는 서명된 URL과 마찬가지로 URL에 정책 설명을 포함하지 않습니다. 정책 설명을 만들려면 다음 절차를 수행하십시오.

미리 준비된 정책을 사용하는 서명된 URL에 대한 정책 설명을 만들려면

1. 다음 JSON 형식과 UTF-8 문자 인코딩을 사용하여 정책 설명을 구성합니다. 지정된 모든 문장 부호 및 기타 리터럴 값을 정확히 포함해야 합니다. `Resource` 및 `DateLessThan` 파라미터에 대한 내용은 미리 준비된 정책을 사용하는 서명된 URL에 대한 정책 설명에서 지정한 값 (p. 130)를 참조하십시오.

```
{
   "Statement":[
      {
         "Resource":"base URL or stream name",
         "Condition":{
            "DateLessThan":{
               "AWS:EpochTime":ending date and time in Unix time format and UTC
            }
         }
      }
   ]
}
```

2. 정책 설명에서 모든 공백(탭과 줄바꿈 문자 포함)을 제거합니다. 애플리케이션 코드의 문자열에 이스케이프 문자를 포함해야 할 수도 있습니다.

미리 준비된 정책을 사용하는 서명된 URL에 대한 정책 설명에서 지정한 값

미리 준비된 정책에 대한 정책 설명을 만들 때 다음 값을 지정합니다.

Resource

지정하는 값은 웹 배포 또는 RTMP 배포 중 어떤 것에 대해 서명된 URL을 만드느냐에 따라 달라집니다.

> **Note**
>
> `Resource`에 대해 값을 하나만 지정할 수 있습니다.

웹 배포

쿼리 문자열을 포함하고(있는 경우) 다음과 같은 CloudFront `Expires`, `Signature` 및 `Key-Pair-Id` 파라미터를 제외한 기본 URL:

```
http://d111111abcdef8.cloudfront.net/images/horizon.jpg?
size=large&license=yes
```

다음을 참조하십시오.

- 프로토콜 – 이 값은 `http://` 또는 `https://`로 시작해야 합니다.
- 쿼리 문자열 파라미터 – 쿼리 문자열 파라미터가 없는 경우, 물음표를 생략합니다.
- 대체 도메인 이름 – URL에 대체 도메인 이름(CNAME)을 지정하는 경우, 웹 페이지 또는 애플리케이션의 파일을 참조할 때 대체 도메인 이름을 지정해야 합니다. 객체에 대한 Amazon S3 URL을 지정하지 마십시오.

RTMP 배포

스트림 이름만 포함하십시오. 예를 들면 스트리밍 비디오의 전체 URL이 다음과 같은 경우

```
rtmp://s5c39gqb8ow64r.cloudfront.net/videos/cfx/st/mp3_name.mp3
```

`Resource`에 다음 값을 사용합니다.

```
videos/mp3_name
```

`mp3:` 또는 `mp4:`와 같은 접두사를 포함하지 마세요. 또한 사용하는 플레이어에 따라 `Resource`의 값에서 파일 확장명을 생략해야 할 수 있습니다. 예를 들면 `sydney-vacation` 대신 `sydney-vacation.flv`을 사용해야 할 수도 있습니다.

DateLessThan

Unix 시간 형식(초) 및 협정 세계시(UTC) 기준의 URL 만료 날짜 및 시간. 예를 들면 2013년 1월 1일 오전 10시(UTC)를 Unix 시간 형식의 1357034400으로 변환합니다.

이 값은 서명된 URL의 `Expires` 쿼리 문자열 파라미터 값과 일치해야 합니다. 값을 인용 부호로 묶지 마세요.

자세한 정보는 CloudFront가 서명된 URL에서 만료 날짜 및 시간을 확인하는 경우 (p. 126) 단원을 참조하십시오.

미리 준비된 정책을 사용하는 서명된 URL에 대한 정책 설명 예제

서명된 URL에 다음 예제의 정책 설명을 사용할 경우, 사용자는 UTC 기준 2013년 1월 1일 오전 10시까지 `http://d111111abcdef8.cloudfront.net/horizon.jpg` 파일에 액세스할 수 있습니다.

```
{
   "Statement":[
      {
         "Resource":"http://d111111abcdef8.cloudfront.net/horizon.jpg?
size=large&license=yes",
         "Condition":{
            "DateLessThan":{
               "AWS:EpochTime":1357034400
            }
         }
      }
   ]
}
```

미리 준비된 정책을 사용하는 서명된 URL에 대한 서명 생성

서명된 URL의 `Signature` 파라미터 값을 만들려면 미리 준비된 정책을 사용하는 서명된 URL의 정책 설명 생성 (p. 129)에서 만든 정책 설명을 해시하고 서명합니다. 이 절차에는 두 가지 버전이 있습니다. 시나리오에 대해 다음 절차를 따릅니다.

- 옵션 1: 미리 준비된 정책을 사용하여 Adobe Flash Player 없이 웹 배포 또는 RTMP 배포에 대한 서명을 만들려면 (p. 131)
- 옵션 2: 미리 준비된 정책을 사용하여 RTMP 배포에 대한 서명을 만들려면(Adobe Flash Player 사용) (p. 132)

다음을 참조하여 정책 설명을 해시, 서명 및 인코딩하는 방법에 대한 추가적인 내용과 예제를 확인하세요.

- Base64 인코딩 및 암호화를 위한 Linux 명령 및 OpenSSL 사용 (p. 155)
- 서명 URL에 대한 서명을 만드는 코드 예제 (p. 156)
- 비공개 콘텐츠 구성을 위한 도구 및 코드 예제 (p. 454)

옵션 1: 미리 준비된 정책을 사용하여 Adobe Flash Player 없이 웹 배포 또는 RTMP 배포에 대한 서명을 만들려면

1. SHA-1 해시 함수 및 RSA를 사용하여 미리 준비된 정책을 사용하는 서명된 URL에 대한 정책 설명을 만들려면 (p. 130) 절차에서 만든 RSA 정책 설명을 해시하고 서명합니다. 공백이 삭제된 버전의 정책 설명을 사용하세요.

   해시 함수에 필요한 프라이빗 키의 경우, 신뢰할 수 있는 활성 서명자와 연결된 프라이빗 키를 사용하십시오.

Note

정책 설명을 해시 및 서명하는 방법은 프로그래밍 언어와 플랫폼에 따라 달라집니다. 샘플 코드에 대한 내용은 서명 URL에 대한 서명을 만드는 코드 예제 (p. 156)를 참조하십시오.

2. 해시 및 서명된 문자열에서 모든 공백(탭과 줄바꿈 문자 포함)을 제거합니다.
3. MIME base64 인코딩 기준으로 문자열을 base64로 인코딩합니다. 자세한 내용은 RFC 2045, MIME(Multipurpose Internet Mail Extensions) Part One: Format of Internet Message Bodies의 Section 6.8, Base64 Content-Transfer-Encoding을 참조하십시오.
4. URL 쿼리 문자열에서 사용할 수 없는 문자를 유효한 문자로 교체합니다. 아래 표에 사용할 수 없는 문자와 유효한 문자가 나열되어 있습니다.

무효 문자를	교체할 유효한 문자
+	- (하이픈)
=	_ (밑줄)
/	~ (물결표)

5. 서명된 URL의 `&Signature=` 뒤에 결과 값을 추가하고 미리 준비된 정책을 사용하여 서명된 URL을 만들려면 (p. 127)로 돌아가서 서명된 URL 부분을 마저 연결합니다.

옵션 2: 미리 준비된 정책을 사용하여 RTMP 배포에 대한 서명을 만들려면(Adobe Flash Player 사용)

1. SHA-1 해시 함수 및 RSA를 사용하여 미리 준비된 정책을 사용하는 서명된 URL에 대한 정책 설명을 만들려면 (p. 130) 절차에서 만든 RSA 정책 설명을 해시하고 서명합니다. 공백이 삭제된 버전의 정책 설명을 사용하세요.

   해시 함수에 필요한 프라이빗 키의 경우, 신뢰할 수 있는 활성 서명자와 연결된 프라이빗 키를 사용하십시오.

   Note

   정책 설명을 해시 및 서명하는 방법은 프로그래밍 언어와 플랫폼에 따라 달라집니다. 샘플 코드에 대한 내용은 서명 URL에 대한 서명을 만드는 코드 예제 (p. 156)를 참조하십시오.
2. 해시 및 서명된 문자열에서 모든 공백(탭과 줄바꿈 문자 포함)을 제거합니다.

   Adobe Flash Player를 사용하고 웹 페이지에서 스트림 이름을 전달 받은 경우, 3단계로 계속합니다.

   Adobe Flash Player를 사용 중인데 웹 페이지에서 스트림 이름이 전달되지 않은 경우, 이 절차의 나머지 부분을 건너뜁니다. 예를 들면 Adobe Flash .swf 파일에서 스트림 이름을 가져오는 자체 플레이어를 개발한 경우, 나머지 절차를 건너뜁니다.
3. MIME base64 인코딩 기준으로 문자열을 base64로 인코딩합니다. 자세한 내용은 RFC 2045, MIME(Multipurpose Internet Mail Extensions) Part One: Format of Internet Message Bodies의 Section 6.8, Base64 Content-Transfer-Encoding을 참조하십시오.
4. URL 쿼리 문자열에서 사용할 수 없는 문자를 유효한 문자로 교체합니다. 아래 표에 사용할 수 없는 문자와 유효한 문자가 나열되어 있습니다.

무효 문자를	교체할 유효한 문자
+	- (하이픈)
=	_ (밑줄)

무효 문자를	교체할 유효한 문자
/	~ (물결표)

5. Adobe Flash Player의 일부 버전에서는 ?, =, & 문자를 URL로 인코딩해야 합니다. 문자를 대체해야 하는 Adobe Flash Player 버전에 대한 내용은 Adobe 웹 사이트를 참조하십시오.

   Flash 버전에서 이러한 문자의 URL 인코딩이 필요없는 경우, 6단계로 건너뜁니다.

   Flash 버전에서 이러한 문자의 URL 인코딩이 필요한 경우, 다음 테이블에 지정된 대로 교체합니다. 이미 이전 단계에서 =를 교체했습니다.

무효 문자를	URL 인코딩으로 교체
?	%3F
&	%26

6. 서명된 URL의 `&Signature=` 뒤에 결과 값을 추가하고 미리 준비된 정책을 사용하여 서명된 URL을 만들려면 (p. 127)로 돌아가서 서명된 URL 부분을 마저 연결합니다.

사용자 지정 정책을 사용한 서명된 URL 생성

주제

- 사용자 지정 정책을 사용하는 서명된 URL에 대한 정책 설명 생성 (p. 136)
- 사용자 지정 정책을 사용하는 서명된 URL에 대한 예제 정책 설명 (p. 138)
- 사용자 지정 정책을 사용하는 서명된 URL에 대한 서명 생성 (p. 140)

사용자 지정 정책을 사용하여 서명된 URL을 만들려면 다음 절차를 수행합니다.

사용자 지정 정책을 사용하여 서명된 URL을 만들려면

1. .NET 또는 Java를 사용하여 서명된 URL을 만드는 중인데 키 페어의 프라이빗 키를 기본 .pem 형식에서 .NET 또는 Java와 상환되는 형식으로 다시 포맷하지 않았다면 지금 포맷하세요. 자세한 내용은 CloudFront 프라이빗 키 재포맷(.NET 및 Java만 해당) (p. 121) 단원을 참조하십시오.
2. 지정된 순서대로 다음 값을 연결하고 사이의 공백(탭과 줄바꿈 문자 포함)을 제거합니다. 애플리케이션 코드의 문자열에 이스케이프 문자를 포함해야 할 수도 있습니다. 모든 값에는 일종의 문자열이 있습니다. 다음 두 가지 예처럼 각 부분을 숫자(1)로 입력합니다.

   1 ### ## URL

   기본 URL이란 서명된 URL을 사용하지 않고 파일에 액세스할 때 사용할 CloudFront URL을 말합니다. 자체 쿼리 문자열 파라미터가 있는 경우 여기에 포함됩니다. 웹 배포의 URL 형식에 대한 자세한 내용은 CloudFront에서 파일에 대한 URL 형식 사용자 지정 (p. 79)을 참조하십시오.

   다음 예는 웹 배포에 대해 지정하는 값을 보여줍니다.

   - 다음 CloudFront URL은 웹 배포의 파일을 위한 것입니다(CloudFront 도메인 이름 사용). `image.jpg`는 `images` 디렉터리에 있습니다. URL 파일의 경로는 HTTP 서버 또는 Amazon S3 버킷의 파일 경로와 일치해야 합니다.

     ```
     http://d111111abcdef8.cloudfront.net/images/image.jpg
     ```

   - 다음 CloudFront URL은 쿼리 문자열을 포함합니다.

     ```
     http://d111111abcdef8.cloudfront.net/images/image.jpg?size=large
     ```

- 다음 CloudFront URL은 웹 배포의 파일을 위한 것입니다. 둘 다 대체 도메인 이름을 사용하며 두 번째는 쿼리 문자열을 포함합니다.

  ```
  http://www.example.com/images/image.jpg
  ```

  ```
  http://www.example.com/images/image.jpg?color=red
  ```

- 다음 CloudFront URL은 대체 도메인 이름과 HTTPS 프로토콜을 사용하는 웹 배포의 파일을 위한 것입니다.

  ```
  https://www.example.com/images/image.jpg
  ```

RTMP 배포의 경우, 다음은 MP4 및 FLV라는 두 가지 비디오 형식의 파일에 대한 예제입니다.

- MP4 – `mp4:sydney-vacation.mp4`
- FLV – `sydney-vacation`
- FLV – `sydney-vacation.flv`

> **Note**
>
> .flv 파일의 경우 플레이어에 따라 `.flv` 파일 이름 확장자의 포함 여부가 결정됩니다. MP3 오디오 파일 또는 H.264/MPEG-4 비디오 파일을 제공하려면 파일 이름에 `mp3:` 또는 `mp4:` 접두사를 적용해야 할 수 있습니다. 일부 미디어 플레이어에서는 자동으로 해당 접두사가 추가되도록 구성되어 있습니다. 그러한 미디어 플레이어에서는 또한 파일 확장명 없이(예: sydney-vacation.mp4 대신 sydney-vacation) 파일 이름을 지정하도록 요구할 수 있습니다.

2 ?

The ?는 기본 URL 뒤에 쿼리 문자열 파라미터가 있음을 나타냅니다. 자체 쿼리 문자열 파라미터가 없는 경우에는 ?를 포함하십시오.

3 ***## ## ### ####(## ##)***&

이 값은 선택 사항입니다. 예를 들어, 자체 쿼리 문자열 파라미터로

```
color=red&size=medium
```

을 추가하고 싶은 경우 ? 뒤(**2** 참조)와 `Policy` 파라미터 앞 사이에 추가합니다. 드물지만 `Key-Pair-Id` 뒤에 쿼리 문자열 파라미터를 넣어야 하는 경우도 있습니다.

> **Important**
>
> 파라미터는 `Policy`, `Signature` 또는 `Key-Pair-Id`라는 이름으로 지정할 수 없습니다.

자체 파라미터를 추가하는 경우, 마지막 파라미터를 포함하여 각 파라미터 뒤에 &를 추가합니다.

4 **Policy=*## ### base64 ### ##***

공백이 제거된 JSON 형식의 정책 설명은 base64로 인코딩합니다. 자세한 내용은 사용자 지정 정책을 사용하는 서명된 URL에 대한 정책 설명 생성 (p. 136) 단원을 참조하십시오.

정책 설명은 사용자에게 부여한 액세스를 제어하는데 이는 파일의 URL(웹 배포) 또는 스트림 이름(RTMP 배포), 만료 날짜 및 시간, URL이 유효한 선택적 날짜 및 시간, 파일을 액세스 할 수 있는 선택적 IP 주소 또는 IP 주소 범위 등이 있습니다.

5 **&Signature=*## ### ## # ### ##***

JSON 정책 설명을 해시, 서명 및 base64로 인코딩한 버전입니다. 자세한 내용은 사용자 지정 정책을 사용하는 서명된 URL에 대한 서명 생성 (p. 140) 단원을 참조하십시오.

❻ **&Key-Pair-Id=*## ### ### # #### # ### ## CloudFront # ## Id***

APKA9ONS7QCOWEXAMPLE와 같은 활성 CloudFront 키 페어의 ID. CloudFront는 서명된 URL을 확인할 때 사용할 퍼블릭 키를 CloudFront 키 페어 ID로 판단합니다. CloudFront는 서명의 정보를 정책 설명의 정보와 비교하고 URL이 변조되지 않았음을 확인합니다.

CloudFront 서명된 URL에 포함한 키 페어 ID는 신뢰할 수 있는 서명자 중 하나의 활성 키 페어 ID여야 합니다.

- 웹 배포 – 키 페어는 캐시 동작의 신뢰할 수 있는 서명자 중 하나인 AWS 계정과 연결되어야 합니다.
- RTMP 배포 – 키 페어는 배포에 대한 신뢰할 수 있는 서명자 중 하나인 AWS 계정과 연결되어야 합니다.

자세한 내용은 서명된 URL 및 서명된 쿠키(신뢰할 수 있는 서명자)를 만들 수 있는 AWS 계정 지정 (p. 118) 단원을 참조하십시오.

CloudFront 키 페어를 교체하면서 키 페어 하나를 비활성화하고 프로그래밍 방식으로 서명된 URL을 만들려는 경우, 신뢰할 수 있는 서명자 중 하나의 새로운 활성 키 페어를 사용하도록 애플리케이션을 업데이트해야 합니다. 서명된 URL을 수동으로 만들 때는 새로 만들어야 합니다. 키 페어 교체에 대한 자세한 내용은 CloudFront 키 페어 교체 (p. 123)를 참조하십시오.

웹 배포의 서명된 URL 예제:

❶ **http://d111111abcdef8.cloudfront.net/image.jpg** ❷ **?**
❸ **color=red&size=medium&**
❹ **Policy=eyANCiAgICEXAMPLEW1lbnQiOiBbeyANCiAgICAgICJSZXNvdXJjZSI6Imh0dHA6Ly9kemJlc3FtN3VuMW0wLmNsb3VkZnJvbnQubmV0L2RlbW8ucGhwIiwgDQogICAgICAiQ29uZGl0aW9uIjp7IA0KICAgICAgICAgIklwQWRkcmVzcyI6eyJBV1M6U291cmNlSXAiOiIyMDcuMTcxLjE4MC4xMDEvMzIifSwNCiAgICAgICAgICJEYXRlR3JlYXRlclRoYW4iOnsiQVdTOkVwb2NoVGltZSI6MTI5Njg2MDE3Nn0sDQogICAgICAgICAiRGF0ZUxlc3NUaGFuIjp7IkFXUzpFcG9jaFRpbWUiOjEyOTY4NjAyMjZ9DQogICAgICB9IA0KICAgfV0gDQp9DQo**
❺ **&Signature=nitfHRCrtziwO2HwPfWw~yYDhUF5EwRunQA-j19DzZrvDh6hQ73lDx~-ar3UocvvRQVw6EkC~GdpGQyyOSKQim-TxAnW7d8F5Kkai9HVx0FIu-5jcQb0UEmat**
EXAMPLE3ReXySpLSMj0yCd3ZAB4UcBCAqEijkytL6f3fVYNGQI6 ❻ **&Key-Pair-Id=APKA9ONS7QCOWEXAMPLE**

RTMP 배포의 서명된 URL 예제:

❶ **videos/mediafile.flv** ❷ **?** ❸ **color=red&size=medium&**
❹ **Policy=eyANCiAgICEXAMPLEW1lbnQiOiBbeyANCiAgICAgICJSZXNvdXJjZSI6Imh0dHA6Ly9kemJlc3FtN3VuMW0wLmNsb3VkZnJvbnQubmV0L2RlbW8ucGhwIiwgDQogICAgICAiQ29uZGl0aW9uIjp7IA0KICAgICAgICAgIklwQWRkcmVzcyI6eyJBV1M6U291cmNlSXAiOiIyMDcuMTcxLjE4MC4xMDEvMzIifSwNCiAgICAgICAgICJEYXRlR3JlYXRlclRoYW4iOnsiQVdTOkVwb2NoVGltZSI6MTI5Njg2MDE3Nn0sDQogICAgICAgICAiRGF0ZUxlc3NUaGFuIjp7IkFXUzpFcG9jaFRpbWUiOjEyOTY4NjAyMjZ9DQogICAgICB9IA0KICAgfV0gDQp9DQo**
❺ **&Signature=nitfHRCrtziwO2HwPfWw~yYDhUF5EwRunQA-j19DzZrvDh6hQ73lDx~-ar3UocvvRQVw6EkC~GdpGQyyOSKQim-TxAnW7d8F5Kkai9HVx0FIu-5jcQb0UEmat**
EXAMPLE3ReXySpLSMj0yCd3ZAB4UcBCAqEijkytL6f3fVYNGQI6 ❻ **&Key-Pair-Id=APKA9ONS7QCOWEXAMPLE**

사용자 지정 정책을 사용하는 서명된 URL에 대한 정책 설명 생성

사용자 지정 정책에 대한 정책 설명을 만들려면 다음 절차를 수행하십시오. 다양한 방법으로 파일에 대한 액세스를 제어하는 몇 가지 예제 정책 설명에 대한 내용은 사용자 지정 정책을 사용하는 서명된 URL에 대한 예제 정책 설명 (p. 138) 단원을 참조하십시오.

사용자 지정 정책을 사용하는 서명된 URL에 대한 정책 설명을 만들려면

1. 다음 JSON 형식을 사용하여 정책 설명을 구성하세요. 자세한 내용은 사용자 지정 정책을 사용하는 서명된 URL에 대한 정책 설명에서 지정한 값 (p. 137) 단원을 참조하십시오.

```
{
   "Statement": [
      {
         "Resource":"URL or stream name of the file",
         "Condition":{
            "DateLessThan":{"AWS:EpochTime":required ending date and time in Unix time
 format and UTC},
            "DateGreaterThan":{"AWS:EpochTime":optional beginning date and time in Unix
 time format and UTC},
            "IpAddress":{"AWS:SourceIp":"optional IP address"}
         }
      }
   ]
}
```

   다음을 참조하십시오.

   - 단 한 개의 명령문만 포함시킬 수 있습니다.
   - UTF-8 문자 인코딩을 사용하세요.
   - 지정된 대로 정확하게 모든 문장 부호 및 파라미터 이름을 포함하세요. 파라미터 이름의 약어는 허용되지 않습니다.
   - `Condition` 섹션의 파라미터 순서는 중요하지 않습니다.
   - `Resource`, `DateLessThan`, `DateGreaterThan`, `IpAddress`에 대한 값 관련 내용은 사용자 지정 정책을 사용하는 서명된 URL에 대한 정책 설명에서 지정한 값 (p. 137)을 참조하십시오.
2. 정책 설명에서 모든 공백(탭과 줄바꿈 문자 포함)을 제거합니다. 애플리케이션 코드의 문자열에 이스케이프 문자를 포함해야 할 수도 있습니다.
3. MIME base64 인코딩 기준으로 정책 설명을 Base64로 인코딩합니다. 자세한 내용은 RFC 2045, MIME(Multipurpose Internet Mail Extensions) Part One: Format of Internet Message Bodies의 Section 6.8, Base64 Content-Transfer-Encoding을 참조하십시오.
4. URL 쿼리 문자열에서 사용할 수 없는 문자를 유효한 문자로 교체합니다. 아래 표에 사용할 수 없는 문자와 유효한 문자가 나열되어 있습니다.

무효 문자를	교체할 유효한 문자
+	- (하이픈)
=	_ (밑줄)
/	~ (물결표)

5. 서명된 URL의 `Policy=` 뒤에 결과 값을 추가하세요.
6. 정책 설명을 해시, 서명 및 base64로 인코딩하여 서명된 URL에 대한 서명을 만드세요. 자세한 내용은 사용자 지정 정책을 사용하는 서명된 URL에 대한 서명 생성 (p. 140) 단원을 참조하십시오.

사용자 지정 정책을 사용하는 서명된 URL에 대한 정책 설명에서 지정한 값

사용자 지정 정책에 대한 정책 설명을 생성할 때 다음 값을 지정합니다.

리소스

지정하는 값은 웹 또는 RTMP 배포 중 어떤 것에 대하여 서명된 URL을 만드느냐에 따라 달라집니다.

> **Note**
>
> `Resource`에 대해 값을 하나만 지정할 수 있습니다.

웹 배포(선택 사항이지만 권장됨)

쿼리 문자열을 포함하고(있는 경우) 다음과 같은 CloudFront `Policy`, `Signature` 및 `Key-Pair-Id` 파라미터를 제외한 기본 URL:

```
http://d111111abcdef8.cloudfront.net/images/horizon.jpg?
size=large&license=yes
```

> **Important**
>
> 웹 배포에서 Resource 파라미터를 생략하는 경우, 서명된 URL을 만들 때 사용하는 키 페어와 연결된 모든 배포 관련 파일에 사용자가 액세스할 수 있게 됩니다.

다음을 참조하십시오.

- 프로토콜 – 이 값은 `http://`, `https://` 또는 `*`으로 시작해야 합니다.
- 쿼리 문자열 파라미터 – 쿼리 문자열 파라미터가 없는 경우, 물음표를 생략합니다.
- 와일드카드 문자 – 0개 이상의 문자(*)에 해당하거나 문자열 내 정확히 1문자(?)에 해당하는 와일드카드를 문자열 내 모든 곳에 사용할 수 있습니다. 예를 들어 값은

  ```
  http://d111111abcdef8.cloudfront.net/*game_download.zip*
  ```

  다음 파일을 포함합니다.

  - `http://d111111abcdef8.cloudfront.net/game_download.zip`
  - `http://d111111abcdef8.cloudfront.net/example_game_download.zip?license=yes`
  - `http://d111111abcdef8.cloudfront.net/test_game_download.zip?license=temp`
- 대체 도메인 이름 – URL에 대체 도메인 이름(CNAME)을 지정하는 경우, 웹 페이지 또는 애플리케이션의 파일을 참조할 때 대체 도메인 이름을 지정해야 합니다. 파일에 대한 Amazon S3 URL을 지정하지 마십시오.

RTMP 배포

스트림 이름만 포함하십시오. 예를 들면 스트리밍 비디오의 전체 URL이 다음과 같은 경우

```
rtmp://s5c39gqb8ow64r.cloudfront.net/videos/cfx/st/mp3_name.mp3
```

`Resource`에 다음 값을 사용합니다.

```
videos/mp3_name
```

`mp3:` 또는 `mp4:`와 같은 접두사를 포함하지 마세요. 또한 사용하는 플레이어에 따라 `Resource`의 값에서 파일 확장명을 생략해야 할 수 있습니다. 예를 들면 `sydney-vacation` 대신 `sydney-vacation.flv`을 사용해야 할 수도 있습니다.

DateLessThan

Unix 시간 형식(초) 및 협정 세계시(UTC) 기준의 URL 만료 날짜 및 시간. 값을 인용 부호로 묶지 마세요. UTC에 대한 자세한 내용은 RFC 3339, 인터넷의 날짜 및 시간: 타임스탬프, http://tools.ietf.org/html/rfc3339를 참조하십시오.

예를 들면 2013년 1월 1일 오전 10시(UTC)를 Unix 시간 형식의 1357034400으로 변환합니다.

이는 `Condition` 섹션에서 유일한 필수 파라미터입니다. CloudFront는 이 값을 통해 프라이빗 콘텐츠에 대한 영구적인 액세스를 방지합니다.

자세한 내용은 CloudFront가 서명된 URL에서 만료 날짜 및 시간을 확인하는 경우 (p. 126) 단원을 참조하십시오.

DateGreaterThan(선택 사항)

Unix 시간 형식(초)과 협정 세계시(UTC)의 URL에 대한 선택적 시작 날짜 및 시간. 지정된 날짜 및 시간 전에 사용자는 파일에 액세스할 수 없습니다. 값을 인용 부호로 묶지 마세요.

IpAddress(선택 사항)

클라이언트의 IP 주소는 GET 요청을 합니다. 다음을 참조하십시오.

- 파일에 액세스하는 IP 주소를 허용하려면 `IpAddress` 파라미터를 생략합니다.
- 하나의 IP 주소 또는 하나의 IP 주소 범위를 지정할 수 있습니다. 예를 들면, 클라이언트의 IP 주소가 두 개의 다른 범위 중 하나에 있는 경우, 정책을 설정하여 액세스를 허용할 수 없습니다.
- 단일 IP 주소에서 액세스를 허용하기 위해 다음을 지정합니다.

  `"IPv4 IP ##/32"`
- IP 주소 범위는 스탠다드 IPv4 CIDR 형식(예: `192.0.2.0/24`)을 지정해야 합니다. 자세한 내용은 RFC 4632, Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan, http://tools.ietf.org/html/rfc4632을 참조하십시오.

  Important

  IPv6 형식의 IP 주소(예: 2001:0db8:85a3:0000:0000:8a2e:0370:7334)는 지원되지 않습니다.

  `IpAddress`를 포함하는 사용자 지정 정책을 사용할 경우 배포에 대해 IPv6를 사용하도록 설정하지 마세요. IP 주소로 일부 콘텐츠에 대한 액세스를 제한하고, 다른 콘텐츠에 대해서는 IPv6 요청을 지원하려면 두 가지 배포를 만들면 됩니다. 자세한 내용은 IPv6 사용 (p. 55) 주제에서 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.

사용자 지정 정책을 사용하는 서명된 URL에 대한 예제 정책 설명

다음 예제 정책 설명은 특정 파일, 디렉터리의 모든 파일 또는 키 페어 ID에 연결된 모든 파일의 액세스 제어 방법을 보여줍니다. 예제는 또한 개별 IP 주소 또는 IP 주소 범위에서 액세스를 제어하는 방법과 지정된 날짜 및 시간 이후 사용자의 서명된 URL 사용 방지 방법을 보여줍니다.

이러한 예제를 복사하여 붙여 넣는 경우, 공백(탭과 줄바꿈 문자 포함)을 제거하고 값을 자체 값과 교체한 후 닫는 괄호(}) 뒤에 줄바꿈 문자를 포함합니다.

자세한 내용은 사용자 지정 정책을 사용하는 서명된 URL에 대한 정책 설명에서 지정한 값 (p. 137) 단원을 참조하십시오.

주제

- 예제 정책 설명: IP 주소 범위에서 하나의 파일에 액세스 (p. 138)
- 예제 정책 설명: IP 주소 범위에서 디렉터리의 모든 파일 액세스 (p. 139)
- 예제 정책 설명: 하나의 IP 주소에서 키 페어 ID에 연결된 모든 파일에 액세스 (p. 139)

예제 정책 설명: IP 주소 범위에서 하나의 파일에 액세스

다음 서명된 URL의 예제 사용자 지정 정책은 UTC 기준 2013년 1월 1일 오전 10시까지 `192.0.2.0/24` 범위의 IP 주소에서 `http://d111111abcdef8.cloudfront.net/game_download.zip` 파일에 액세스할 수 있는 사용자를 지정합니다.

```
{
   "Statement": [
      {
         "Resource":"http://d111111abcdef8.cloudfront.net/game_download.zip",
         "Condition":{
            "IpAddress":{"AWS:SourceIp":"192.0.2.0/24"},
            "DateLessThan":{"AWS:EpochTime":1357034400}
         }
      }
   ]
}
```

예제 정책 설명: IP 주소 범위에서 디렉터리의 모든 파일 액세스

다음 예제 사용자 지정 정책은 `Resource` 파라미터의 * 와일드카드 문자로 표시된 바와 같이 `training` 디렉터리의 파일에 대한 서명된 URL을 만들 수 있도록 허용합니다. 사용자는 UTC 기준 2013년 1월 1일 오전 10시까지 `192.0.2.0/24` 범위의 IP 주소에서 파일에 액세스할 수 있습니다.

```
{
   "Statement": [
      {
         "Resource":"http://d111111abcdef8.cloudfront.net/training/*",
         "Condition":{
            "IpAddress":{"AWS:SourceIp":"192.0.2.0/24"},
            "DateLessThan":{"AWS:EpochTime":1357034400}
         }
      }
   ]
}
```

이 정책을 사용하는 서명된 URL은 각자 특정 파일을 식별하는 다음과 같은 기본 URL을 포함합니다.

`http://d111111abcdef8.cloudfront.net/training/orientation.pdf`

예제 정책 설명: 하나의 IP 주소에서 키 페어 ID에 연결된 모든 파일에 액세스

다음 샘플 사용자 지정 정책은 `Resource`의 * 와일드카드 문자 파라미터에 표시된 대로 배포와 연결된 파일에 대한 서명된 URL을 만들 수 있도록 허용합니다. 사용자는 IP 주소 `192.0.2.10/32`를 사용해야 합니다. `192.0.2.10/32` CIDR 표기법의 값은 단일 IP 주소 `192.0.2.10`를 가리킵니다. 파일은 UTC 기준 2013년 1월 1일 오전 10시에서 2013년 1월 2일 오전 10시까지만 사용할 수 있습니다.

```
{
   "Statement": [
      {
         "Resource":"http://*",
         "Condition":{
            "IpAddress":{"AWS:SourceIp":"192.0.2.10/32"},
            "DateGreaterThan":{"AWS:EpochTime":1357034400},
            "DateLessThan":{"AWS:EpochTime":1357120800}
         }
      }
   ]
}
```

이 정책을 사용하는 서명된 URL은 각자 특정 CloudFront 배포의 특정 파일을 식별하는 다음과 같은 기본 URL을 포함합니다.

`http://d111111abcdef8.cloudfront.net/training/orientation.pdf`

또한 서명된 URL은 키 페어 ID를 포함하고 이는 기본 URL에서 지정한 배포(d111111abcdef8.cloudfront.net)의 신뢰할 수 있는 서명자와 연결되어야 합니다.

사용자 지정 정책을 사용하는 서명된 URL에 대한 서명 생성

서명된 URL에 대한 서명은 정책 설명의 해시, 서명 및 base64로 인코딩된 버전의 사용자 지정 정책을 사용합니다. 사용자 지정 정책에 대한 서명을 만들려면 시나리오에 대한 절차를 따릅니다. 배포 유형(웹 또는 RTMP)에 따라 선택하는 버전이 달라지고 RTMP 배포의 경우, Adobe Flash Player 또는 기타 미디어 플레이어를 사용합니다.

- 옵션 1: 사용자 지정 정책을 사용하여 Adobe Flash Player 없이 웹 배포 또는 RTMP 배포에 대한 서명을 만들려면 (p. 140)
- 옵션 2: 사용자 지정 정책으로 RTMP 배포에 대한 서명을 만들려면(Adobe Flash Player) (p. 140)

다음을 참조하여 정책 설명을 해시, 서명 및 인코딩하는 방법에 대한 추가적인 내용과 예제를 확인하세요.

- Base64 인코딩 및 암호화를 위한 Linux 명령 및 OpenSSL 사용 (p. 155)
- 서명 URL에 대한 서명을 만드는 코드 예제 (p. 156)
- 비공개 콘텐츠 구성을 위한 도구 및 코드 예제 (p. 454)

옵션 1: 사용자 지정 정책을 사용하여 Adobe Flash Player 없이 웹 배포 또는 RTMP 배포에 대한 서명을 만들려면

1. SHA-1 해시 함수 및 RSA를 사용하여 사용자 지정 정책을 사용하는 서명된 URL에 대한 정책 설명을 만들려면 (p. 136) 절차에서 만든 JSON 정책 설명을 해시하고 서명합니다. 더 이상 공백을 포함하지 않지만 base64로 인코딩되지 않은 정책 설명 버전을 사용하세요.

   해시 함수에 필요한 프라이빗 키의 경우, 신뢰할 수 있는 활성 서명자와 연결된 프라이빗 키를 사용하십시오.

   > **Note**
   >
   > 정책 설명을 해시 및 서명하는 방법은 프로그래밍 언어와 플랫폼에 따라 달라집니다. 샘플 코드에 대한 내용은 서명 URL에 대한 서명을 만드는 코드 예제 (p. 156)를 참조하십시오.

2. 해시 및 서명된 문자열에서 모든 공백(탭과 줄바꿈 문자 포함)을 제거합니다.
3. MIME base64 인코딩 기준으로 문자열을 base64로 인코딩합니다. 자세한 내용은 RFC 2045, MIME(Multipurpose Internet Mail Extensions) Part One: Format of Internet Message Bodies의 Section 6.8, Base64 Content-Transfer-Encoding을 참조하십시오.
4. URL 쿼리 문자열에서 사용할 수 없는 문자를 유효한 문자로 교체합니다. 아래 표에 사용할 수 없는 문자와 유효한 문자가 나열되어 있습니다.

무효 문자를	교체할 유효한 문자
+	- (하이픈)
=	_ (밑줄)
/	~ (물결표)

5. 서명된 URL의 `&Signature=` 뒤에 결과 값을 추가하고 사용자 지정 정책을 사용하여 서명된 URL을 만들려면 (p. 133)로 돌아가서 서명된 URL 부분을 마저 연결합니다.

옵션 2: 사용자 지정 정책으로 RTMP 배포에 대한 서명을 만들려면(Adobe Flash Player)

1. SHA-1 해시 함수 및 RSA를 사용하여 사용자 지정 정책을 사용하는 서명된 URL에 대한 정책 설명을 만들려면 (p. 136) 절차에서 만든 JSON 정책 설명을 해시하고 서명합니다. 더 이상 공백을 포함하지 않지만 base64로 인코딩되지 않은 정책 설명 버전을 사용하세요.

해시 함수에 필요한 프라이빗 키의 경우, 신뢰할 수 있는 활성 서명자와 연결된 프라이빗 키를 사용하십시오.

Note

정책 설명을 해시 및 서명하는 방법은 프로그래밍 언어와 플랫폼에 따라 달라집니다. 샘플 코드에 대한 내용은 서명 URL에 대한 서명을 만드는 코드 예제 (p. 156)를 참조하십시오.

2. 해시 및 서명된 문자열에서 모든 공백(탭과 줄바꿈 문자 포함)을 제거합니다.

   웹 페이지에서 스트림 이름을 전달받은 경우, 3단계로 갑니다.

   웹 페이지에서 스트림 이름을 전달받지 않은 경우, 남은 절차를 건너뜁니다. 예를 들면 Adobe Flash .swf 파일에서 스트림 이름을 가져오는 자체 플레이어를 개발한 경우, 나머지 절차를 건너뜁니다.
3. MIME base64 인코딩 기준으로 문자열을 base64로 인코딩합니다. 자세한 내용은 RFC 2045, MIME(Multipurpose Internet Mail Extensions) Part One: Format of Internet Message Bodies의 Section 6.8, Base64 Content-Transfer-Encoding을 참조하십시오.
4. URL 쿼리 문자열에서 사용할 수 없는 문자를 유효한 문자로 교체합니다. 아래 표에 사용할 수 없는 문자와 유효한 문자가 나열되어 있습니다.

무효 문자를	교체할 유효한 문자
+	- (하이픈)
=	_ (밑줄)
/	~ (물결표)

5. Adobe Flash Player의 일부 버전에서는 ?, =, & 문자를 URL로 인코딩해야 합니다. 문자를 대체해야 하는 Adobe Flash Player 버전에 대한 내용은 Adobe 웹 사이트를 참조하십시오.

   Adobe Flash Player 버전에서 ?, =, & 문자를 URL로 인코딩하지 않는 경우, 6단계로 건너뛰세요.

   Adobe Flash Player 버전에서 이러한 문자의 URL 인코딩이 필요한 경우, 다음 표에 따라 교체합니다. 이미 이전 단계에서 =를 교체했습니다.

무효 문자를	URL 인코딩으로 교체
?	%3F
&	%26

6. 서명된 URL의 `&Signature=` 뒤에 결과 값을 추가하고 사용자 지정 정책을 사용하여 서명된 URL을 만들려면 (p. 133)로 돌아가서 서명된 URL 부분을 마저 연결합니다.

서명된 쿠키 사용

현재의 URL을 변경하지 않으려는 경우나 여러 제한된 파일(예: 웹 사이트의 구독자 영역에 있는 전체 파일)에 대한 액세스 권한을 제공하려는 경우, CloudFront서명된 쿠키를 사용하여 콘텐츠 액세스를 제어할 수 있습니다. 이 주제에서는 서명된 쿠키를 사용할 때 고려해야 할 사항과 미리 준비된 정책 및 사용자 지정 정책을 사용하여 서명된 쿠키를 설정하는 방법을 다룹니다.

주제

- 서명된 쿠키에 대하여 미리 준비된(canned) 정책 및 사용자 지정 정책 간 선택 (p. 142)
- 서명된 쿠키 작동 방식 (p. 142)
- 서명된 쿠키 악용 방지 (p. 143)

- CloudFront는 언제 서명된 쿠키의 만료 날짜 및 시간을 확인합니까? (p. 143)
- 샘플 코드 및 타사 도구 (p. 144)
- 미리 준비된 정책을 사용하여 서명된 쿠키 설정 (p. 144)
- 사용자 지정 정책을 사용하여 서명된 쿠키 설정 (p. 148)

서명된 쿠키에 대하여 미리 준비된(canned) 정책 및 사용자 지정 정책 간 선택

서명된 쿠키를 만들 때 JSON 형식의 정책 설명을 작성하여 서명된 쿠키의 제약 조건(예: 쿠키의 유효 기간)을 지정합니다. 미리 준비된 정책 또는 사용자 지정 정책을 사용할 수 있습니다. 다음 테이블은 미리 준비된 정책과 사용자 지정 정책을 비교합니다.

설명	미리 준비된(canned) 정책	사용자 지정 정책
정책 설명은 여러 파일에 재사용할 수 있습니다. 정책 설명을 재사용하려면 `Resource` 객체에 와일드카드 문자를 사용해야 합니다. 자세한 내용은 서명된 쿠키에서 사용자 지정 정책을 위한 정책 설명에서 지정한 값 (p. 152) 단원을 참조하십시오.	아니요	예
날짜 및 시간을 지정하여 사용자가 콘텐츠에 액세스를 시작할 수 있습니다.	아니요	예(선택 사항)
사용자의 콘텐츠 액세스를 중단할 날짜 및 시간을 지정할 수 있습니다.	예	예
콘텐츠에 액세스할 수 있는 사용자의 IP 주소 또는 IP 주소 범위를 지정할 수 있습니다.	아니요	예(선택 사항)

미리 준비된(canned) 정책으로 서명된 쿠키를 만드는 방법에 대한 자세한 내용은 미리 준비된 정책을 사용하여 서명된 쿠키 설정 (p. 144)을 참조하십시오.

사용자 지정 정책으로 서명된 쿠키를 만드는 방법에 대한 자세한 내용은 사용자 지정 정책을 사용하여 서명된 쿠키 설정 (p. 148)을 참조하십시오.

서명된 쿠키 작동 방식

서명된 쿠키에 대한 CloudFront 구성 방법과 사용자가 서명된 쿠키를 포함한 요청을 제출할 때 CloudFront 반응에 대한 개요가 있습니다.

1. CloudFront 배포에서 신뢰할 수 있는 서명자(서명된 URL과 서명된 쿠키를 만들 권한이 있는 AWS 계정)를 하나 이상 지정하십시오.

   자세한 내용은 서명된 URL 및 서명된 쿠키(신뢰할 수 있는 서명자)를 만들 수 있는 AWS 계정 지정 (p. 118) 단원을 참조하십시오.
2. 사용자가 콘텐츠에 액세스해야 하는지 여부를 파악하고, 필요하면 최종 사용자에게 `Set-Cookie` 헤더 세 개를 보내도록 애플리케이션을 개발합니다. (각 `Set-Cookie` 헤더는 이름-값 페어를 하나만 포함할 수 있고 CloudFront 서명된 쿠키에는 이름-값 페어 세 개가 필요합니다.) 최종 사용자가 프라이빗 콘텐츠를 요청하기 전에 `Set-Cookie` 헤더를 최종 사용자에게 보내야 합니다. 쿠키의 만료 시간을 짧게 설정한 경우, 사용자가 계속 액세스할 수 있도록 후속 요청에 대해 `Set-Cookie` 헤더 세 개를 더 보낼 수 있습니다.

일반적으로, CloudFront 배포에는 캐시 동작이 두 개 이상 있는데 하나는 인증이 필요 없고 다른 하나는 필요합니다. 사이트 보안에 대한 오류 페이지에는 로그인 페이지로 가는 리디렉터 또는 링크가 있습니다.

쿠키를 기반으로 파일을 캐시하도록 배포를 구성하는 경우, CloudFront은(는) 서명된 쿠키의 속성을 기반으로 별도의 파일을 캐시하지 않습니다.

3. 사용자가 웹 사이트에 로그인하여 콘텐츠에 대한 비용을 지불하거나 액세스에 필요한 자격 요건을 충족합니다.
4. 애플리케이션은 응답을 통해 `Set-Cookie` 헤더를 반환하고 최종 사용자는 이름-값 페어를 저장합니다.
5. 사용자가 파일을 요청합니다.

   사용자의 브라우저 또는 기타 최종 사용자는 4단계의 이름-값 페어를 불러와서 이를 `Cookie` 헤더의 요청에 추가합니다. 이것이 서명된 쿠키입니다.
6. CloudFront는 퍼블릭 키를 사용하여 서명된 쿠키의 서명을 확인하고 쿠키가 아직 변조되지 않았음을 확인합니다. 유효하지 않은 서명인 경우 요청을 거부합니다.

   쿠키의 서명이 유효한 경우, CloudFront는 쿠키의 정책 설명을 보거나 미리 준비된(canned) 정책을 사용하는 경우, 정책 설명을 구성하여 요청이 아직 유효함을 확인합니다. 예를 들어 쿠키의 시작 및 종료 날짜와 시간을 지정하는 경우, CloudFront는 액세스가 허용된 시간 동안 콘텐츠 액세스를 시도한 사용자를 확인합니다.

   요청이 정책 설명의 요구 사항을 충족하는 경우, CloudFront은(는) 제한되지 않은 콘텐츠와 같은 방식으로 콘텐츠를 제공합니다. 다시 말해 파일이 이미 엣지 캐시에 있는지 여부를 판단하고, 필요에 따라 요청을 오리진으로 전달하고, 사용자에게 파일을 반환합니다.

서명된 쿠키 악용 방지

`Domain` 헤더에 `Set-Cookie` 파라미터를 지정할 때는 루트 도메인 이름을 가진 사용자의 액세스 가능성을 줄이도록 최대한 정확한 값을 지정하십시오. 예를 들자면 example.com보다 app.example.com이 더 좋고, example.com을 제어하지 않을 때는 특히 그렇습니다. 이렇게 하면 www.example.com을 통한 콘텐츠 액세스를 방지할 수 있습니다.

이런 유형의 공격을 방지하려면 다음을 수행하십시오.

- `Expires` 및 `Max-Age` 쿠키 속성을 제외하여 `Set-Cookie` 헤더에서 세션 쿠키를 만들도록 합니다. 세션 쿠키는 사용자가 브라우저를 닫을 때 자동으로 삭제되며, 따라서 권한 없는 제3자가 콘텐츠에 액세스할 가능성을 줄여 줍니다.
- `Secure` 속성을 포함하여 최종 사용자가 이를 요청할 때 쿠키가 암호화되도록 하세요.
- 가능하면 사용자 지정 정책을 사용하고 최종 사용자의 IP 주소를 포함합니다.
- `CloudFront-Expires` 속성에서 사용자의 콘텐츠 액세스를 허용할 기간을 기준으로 하여 가장 짧고 합리적인 만료 시간을 지정하세요.

CloudFront는 언제 서명된 쿠키의 만료 날짜 및 시간을 확인합니까?

CloudFront는 서명된 쿠키가 아직 유효한지 파악하기 위해 HTTP 요청 시 쿠키의 만료 날짜 및 시간을 확인합니다. 클라이언트가 만료 시간 직전에 대용량 파일을 다운로드하기 시작한 경우, 다운로드 도중 만료 시간이 지나도 다운로드는 완료됩니다. TCP 연결이 끊어진 경우, 클라이언트가 만료 시간 이후에 다운로드를 다시 시작하는 것은 불가능합니다.

클라이언트가 범위 GET을 사용하여 파일을 작은 조각으로 가져오는 경우, 만료 시간 이후의 GET 요청은 실패합니다. 범위 GET에 대한 자세한 내용은 CloudFront에서 객체에 대한 부분적인 요청을 처리하는 방법 (Range GET) (p. 214)을 참조하십시오.

샘플 코드 및 타사 도구

프라이빗 콘텐츠에 대한 샘플 코드는 서명된 URL에 대한 서명을 만드는 방법만 보여줍니다. 그러나 서명된 쿠키에 대한 서명 생성 프로세스도 이와 유사하기 때문에 대부분의 샘플 코드는 여전히 관계가 있습니다. 자세한 내용은 다음 주제를 참조하십시오.

- Perl을 사용한 URL 서명 생성 (p. 156)
- PHP를 사용한 URL 서명 생성 (p. 164)
- C# 및 .NET Framework를 사용한 URL 서명 생성 (p. 169)
- Java를 사용한 URL 서명 생성 (p. 174)

서명된 URL 생성 등 프라이빗 콘텐츠를 지원하는 타사 도구에 대한 자세한 내용은 비공개 콘텐츠 구성을 위한 도구 및 코드 예제 (p. 454)을 참조하십시오.

미리 준비된 정책을 사용하여 서명된 쿠키 설정

미리 준비된 정책을 사용하여 서명된 쿠키를 설정하려면 다음 단계를 완료하십시오. 서명을 생성하려면 미리 준비된 정책을 사용하는 서명된 쿠키에 대한 서명 생성 단원을 참조하십시오.

미리 준비된 정책을 사용하여 서명된 쿠키를 설정하려면

1. .NET 또는 Java를 사용하여 서명된 쿠키를 생성하는 중인데 키 페어의 프라이빗 키를 기본 .pem 형식에서 .NET 또는 Java와 상환되는 형식으로 다시 포맷하지 않았다면 지금 포맷하십시오. 자세한 내용은 CloudFront 프라이빗 키 재포맷(.NET 및 Java만 해당) (p. 121) 단원을 참조하십시오.
2. `Set-Cookie` 헤더 세 개를 승인된 최종 사용자에게 보내도록 애플리케이션을 프로그래밍합니다. 각 `Set-Cookie` 헤더는 이름-값 페어를 하나만 포함할 수 있고 CloudFront 서명된 쿠키에는 이름-값 페어 세 개가 필요하기 때문에 `Set-Cookie` 헤더가 세 개 있어야 합니다. 이름-값 페어는 `CloudFront-Expires`, `CloudFront-Signature`, `CloudFront-Key-Pair-Id`입니다. 액세스가 제한되는 파일에 대해 사용자가 첫 번째 요청을 하려면 최종 사용자에게 값이 있어야 합니다.

   > **Note**
   >
   > 일반적으로 `Expires` 및 `Max-Age` 속성을 제외하는 것이 좋습니다. 이러한 속성을 제외하면 사용자가 브라우저를 닫을 때 쿠키가 삭제되므로 권한 없는 사람이 콘텐츠에 액세스할 가능성이 줄어듭니다. 자세한 내용은 서명된 쿠키 악용 방지 (p. 143) 단원을 참조하십시오.

   쿠키 속성의 이름은 대소문자를 구분합니다.

   줄바꿈은 속성의 가독성을 높이기 위해서만 사용됩니다.

```
Set-Cookie:
CloudFront-Expires=date and time in Unix time format (in seconds) and Coordinated
 Universal Time (UTC);
Domain=optional domain name;
Path=/optional directory path;
Secure;
HttpOnly

Set-Cookie:
CloudFront-Signature=hashed and signed version of the policy statement;
Domain=optional domain name;
Path=/optional directory path;
Secure;
HttpOnly

Set-Cookie:
CloudFront-Key-Pair-Id=active CloudFront key pair Id for the key pair that you are
 using to generate the signature;
Domain=optional domain name;
```

```
Path=/optional directory path;
Secure;
HttpOnly
```

(선택) **Domain**

요청된 파일의 도메인 이름입니다. `Domain` 속성을 지정하지 않는 경우, 기본값은 URL의 도메인 이름이고 이는 하위 도메인이 아닌 지정된 도메인 이름에만 적용됩니다. `Domain` 속성을 지정하는 경우, 하위 도메인에도 적용됩니다. 도메인 이름 앞의 점(예: `Domain=.example.com`)은 선택 사항입니다. 또한 `Domain` 속성을 지정하는 경우, URL의 도메인 이름과 `Domain` 속성의 값이 일치해야 합니다.

CloudFront가 배포에 할당하는 도메인 이름을 d111111abcdef8.cloudfront.net과 같이 지정할 수 있으나, *.cloudfront.net을 도메인 이름으로 지정할 수는 없습니다.

URL에 example.com과 같은 대체 도메인 이름을 사용하려면 `Domain` 속성 지정 여부와 상관없이 대체 도메인 이름을 배포에 추가해야 합니다. 자세한 내용은 Alternate Domain Names (CNAMEs) (p. 51) 주제에서 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.

(선택) **Path**

요청된 파일의 경로입니다. `Path` 속성을 지정하지 않는 경우, 기본값은 URL의 경로입니다.

Secure

요청을 보내기 전 최종 사용자에게 쿠키의 암호화를 요청하세요. 쿠키 속성이 MITM(중간자 공격)을 당하지 않도록 HTTPS 연결을 통해 `Set-Cookie` 헤더를 보내는 것이 좋습니다.

HttpOnly

HTTP 또는 HTTPS 요청에 한해 최종 사용자에게 쿠키를 요구하세요.

CloudFront-Expires

Unix 시간 형식(초) 및 협정 세계시(UTC) 기준의 URL 만료 날짜 및 시간을 지정합니다. 예를 들면 2013년 1월 1일 오전 10시(UTC)를 Unix 시간 형식의 1357034400으로 변환합니다. 에포크(epoch) 시간을 사용하려면 날짜에 대해 32비트 정수를 사용합니다. 이 정수를 2147483647(2038년 1월 19일 03:14:07 UTC)를 넘을 수 없습니다. UTC에 대한 자세한 내용은 RFC 3339, 인터넷의 날짜 및 시간: 타임스탬프, http://tools.ietf.org/html/rfc3339를 참조하십시오.

CloudFront-Signature

JSON 정책 설명의 해시, 서명 및 base64 인코딩 버전입니다. 자세한 내용은 미리 준비된 정책을 사용하는 서명된 쿠키에 대한 서명 생성 (p. 146) 단원을 참조하십시오.

CloudFront-Key-Pair-Id

APKA9ONS7QCOWEXAMPLE와 같은 활성 CloudFront 키 페어의 ID입니다. CloudFront는 서명된 쿠키를 확인할 때 사용할 퍼블릭 키를 CloudFront 키 페어 ID로 판단합니다. CloudFront는 서명의 정보를 정책 설명의 정보와 비교하고 URL이 변조되지 않았음을 확인합니다.

CloudFront 서명된 쿠키에 포함하는 키 페어 ID는 캐시 동작에 대한 신뢰할 수 있는 서명자 중 하나인 AWS 계정과 연결되어야 합니다.

자세한 내용은 서명된 URL 및 서명된 쿠키(신뢰할 수 있는 서명자)를 만들 수 있는 AWS 계정 지정 (p. 118) 단원을 참조하십시오.

CloudFront 키 페어를 교체하는 동안 이를 비활성화하고 애플리케이션을 업데이트하여 신뢰할 수 있는 서명자 중 하나에 해당하는 새로운 활성 키 페어를 사용해야 합니다. 키 페어 교체에 대한 자세한 내용은 CloudFront 키 페어 교체 (p. 123)를 참조하십시오.

다음 예제는 배포와 연결된 도메인 이름을 파일의 URL에 사용할 때 서명된 쿠키 한 개의 `Set-Cookie` 헤더를 보여줍니다.

```
Set-Cookie: CloudFront-Expires=1426500000; Domain=d111111abcdef8.cloudfront.net; Path=/
images/*; Secure; HttpOnly
Set-Cookie: CloudFront-Signature=yXrSIgyQoeE4FBI4eMKF6ho~CA8_;
 Domain=d111111abcdef8.cloudfront.net; Path=/images/*; Secure; HttpOnly
Set-Cookie: CloudFront-Key-Pair-Id=APKA9ONS7QCOWEXAMPLE;
 Domain=d111111abcdef8.cloudfront.net; Path=/images/*; Secure; HttpOnly
```

다음 예제는 파일에 대한 URL의 example.org 대체 도메인 이름을 사용할 때 하나의 서명된 쿠키에 대한 예제 `Set-Cookie` 헤더를 보여줍니다.

```
Set-Cookie: CloudFront-Expires=1426500000; Domain=example.org; Path=/images/*; Secure;
 HttpOnly
Set-Cookie: CloudFront-Signature=yXrSIgyQoeE4FBI4eMKF6ho~CA8_; Domain=example.org; Path=/
images/*; Secure; HttpOnly
Set-Cookie: CloudFront-Key-Pair-Id=APKA9ONS7QCOWEXAMPLE; Domain=example.org; Path=/images/
*; Secure; HttpOnly
```

URL에 example.com과 같은 대체 도메인 이름을 사용하려면 `Domain` 속성 지정 여부와 상관없이 대체 도메인 이름을 배포에 추가해야 합니다. 자세한 내용은 Alternate Domain Names (CNAMEs) (p. 51) 주제에서 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.

미리 준비된 정책을 사용하는 서명된 쿠키에 대한 서명 생성

미리 준비된 정책을 사용하는 서명된 쿠키에 대한 서명을 만들려면 다음을 수행하십시오.

1. 정책 설명을 만듭니다. 미리 준비된 정책을 사용하는 서명된 쿠키에 대한 정책 설명 생성 (p. 146) 단원을 참조하십시오.
2. 정책 설명에 서명하여 서명을 만듭니다. 정책 설명에 서명하여 미리 준비된 정책을 사용하는 서명된 쿠키에 대한 서명 생성 (p. 147) 단원을 참조하십시오.

미리 준비된 정책을 사용하는 서명된 쿠키에 대한 정책 설명 생성

미리 준비된 정책을 사용하는 서명된 쿠키를 설정할 때 `CloudFront-Signature` 속성은 정책 설명의 해시 및 서명된 버전입니다. 미리 준비된 정책을 사용하는 서명된 쿠키의 경우, 사용자 지정 정책을 사용하는 서명된 쿠키와 마찬가지로 `Set-Cookie` 헤더에 정책 설명을 포함하지 않습니다. 정책 설명을 만들려면 다음 절차를 수행하십시오.

미리 준비된 정책을 사용하는 서명된 쿠키에 대한 정책 설명을 만들려면

1. 다음 JSON 형식과 UTF-8 문자 인코딩을 사용하여 정책 설명을 구성합니다. 지정된 모든 문장 부호 및 기타 리터럴 값을 정확히 포함해야 합니다. `Resource` 및 `DateLessThan` 파라미터에 대한 내용은 서명된 쿠키에서 미리 준비된 정책을 위한 정책 설명에서 지정한 값 (p. 147)를 참조하십시오.

```
{
   "Statement":[
      {
         "Resource":"base URL or stream name",
         "Condition":{
            "DateLessThan":{
               "AWS:EpochTime":ending date and time in Unix time format and UTC
            }
         }
      }
   ]
}
```

2. 정책 설명에서 모든 공백(탭과 줄바꿈 문자 포함)을 제거합니다. 애플리케이션 코드의 문자열에 이스케이프 문자를 포함해야 할 수도 있습니다.

서명된 쿠키에서 미리 준비된 정책을 위한 정책 설명에서 지정한 값

미리 준비된 정책에 대한 정책 설명을 만들 때 다음 값을 지정합니다.

Resource

쿼리 문자열을 포함하는 기본 URL(있는 경우)은 다음과 같습니다.

```
http://d111111abcdef8.cloudfront.net/images/horizon.jpg?
size=large&license=yes
```

`Resource`에 대해 값을 하나만 지정할 수 있습니다.

다음을 참조하십시오.

- 프로토콜 – 이 값은 `http://` 또는 `https://`로 시작해야 합니다.
- 쿼리 문자열 파라미터 – 쿼리 문자열 파라미터가 없는 경우, 물음표를 생략합니다.
- 대체 도메인 이름 – URL에 대체 도메인 이름(CNAME)을 지정하는 경우, 웹 페이지 또는 애플리케이션의 파일을 참조할 때 대체 도메인 이름을 지정해야 합니다. 파일에 대한 Amazon S3 URL을 지정하지 마십시오.

DateLessThan

Unix 시간 형식(초) 및 협정 세계시(UTC) 기준의 URL 만료 날짜 및 시간. 값을 인용 부호로 묶지 마세요.

예를 들면 2015년 3월 16일 오전 10시(UTC)를 Unix 시간 형식인 1426500000으로 변환합니다.

이 값은 `CloudFront-Expires` 헤더의 `Set-Cookie` 속성 값과 일치해야 합니다. 값을 인용 부호로 묶지 마세요.

자세한 내용은 CloudFront는 언제 서명된 쿠키의 만료 날짜 및 시간을 확인합니까? (p. 143) 단원을 참조하십시오.

미리 준비된 정책에 대한 정책 설명 예제

서명된 쿠키에 다음 예제의 정책 설명을 사용할 경우, 사용자는 UTC 기준 2015년 3월 16일 오전 10시까지 `http://d111111abcdef8.cloudfront.net/horizon.jpg` 파일에 액세스할 수 있습니다.

```
{
   "Statement":[
      {
         "Resource":"http://d111111abcdef8.cloudfront.net/horizon.jpg?
size=large&license=yes",
         "Condition":{
            "DateLessThan":{
               "AWS:EpochTime":1426500000
            }
         }
      }
   ]
}
```

정책 설명에 서명하여 미리 준비된 정책을 사용하는 서명된 쿠키에 대한 서명 생성

`CloudFront-Signature` 헤더의 `Set-Cookie` 속성의 값을 만들려면 미리 준비된 정책을 사용하는 서명된 쿠키에 대한 정책 설명을 만들려면 (p. 146)에서 만든 정책 설명을 해시하고 서명합니다.

다음 주제를 참조하여 정책 설명을 해시, 서명 및 인코딩하는 방법에 대한 추가적인 내용과 예제를 확인하세요.

- Base64 인코딩 및 암호화를 위한 Linux 명령 및 OpenSSL 사용 (p. 155)
- 서명 URL에 대한 서명을 만드는 코드 예제 (p. 156)
- 비공개 콘텐츠 구성을 위한 도구 및 코드 예제 (p. 454)

미리 준비된 정책을 사용하는 서명된 쿠키에 대한 서명을 만들려면

1. SHA-1 해시 함수 및 RSA를 사용하여 미리 준비된 정책을 사용하는 서명된 쿠키에 대한 정책 설명을 만들려면 (p. 146) 절차에서 만든 RSA 정책 설명을 해시하고 서명합니다. 공백이 삭제된 버전의 정책 설명을 사용하세요.

   해시 함수에 필요한 프라이빗 키의 경우, 신뢰할 수 있는 활성 서명자와 연결된 프라이빗 키를 사용하십시오.

   Note

   정책 설명을 해시 및 서명하는 방법은 프로그래밍 언어와 플랫폼에 따라 달라집니다. 샘플 코드에 대한 내용은 서명 URL에 대한 서명을 만드는 코드 예제 (p. 156)를 참조하십시오.
2. 해시 및 서명된 문자열에서 모든 공백(탭과 줄바꿈 문자 포함)을 제거합니다.
3. MIME base64 인코딩 기준으로 문자열을 base64로 인코딩합니다. 자세한 내용은 RFC 2045, MIME(Multipurpose Internet Mail Extensions) Part One: Format of Internet Message Bodies의 Section 6.8, Base64 Content-Transfer-Encoding을 참조하십시오.
4. URL 쿼리 문자열에서 사용할 수 없는 문자를 유효한 문자로 교체합니다. 아래 표에 사용할 수 없는 문자와 유효한 문자가 나열되어 있습니다.

무효 문자를	교체할 유효한 문자
+	- (하이픈)
=	_ (밑줄)
/	~ (물결표)

5. 결과로 얻은 값을 `Set-Cookie` 이름-값 페어의 `CloudFront-Signature` 헤더에 포함하세요. 그런 다음, 미리 준비된 정책을 사용하여 서명된 쿠키를 설정하려면 (p. 144) 절차로 돌아가서 `Set-Cookie`에 대한 `CloudFront-Key-Pair-Id` 헤더를 추가합니다.

사용자 지정 정책을 사용하여 서명된 쿠키 설정

주제

- 사용자 지정 정책을 사용하는 서명된 쿠키에 대한 정책 설명 생성 (p. 151)
- 사용자 지정 정책을 사용하는 서명된 쿠키에 대한 예제 정책 설명 (p. 153)
- 사용자 지정 정책을 사용하는 서명된 쿠키에 대한 서명 생성 (p. 154)

사용자 지정 정책을 사용하는 서명된 쿠키를 설정하려면 다음 절차를 수행합니다.

사용자 지정 정책을 사용하여 서명된 쿠키를 설정하려면

1. .NET 또는 Java를 사용하여 서명된 URL을 만드는 중인데 키 페어의 프라이빗 키를 기본 .pem 형식에서 .NET 또는 Java와 상환되는 형식으로 다시 포맷하지 않았다면 지금 포맷하세요. 자세한 내용은 CloudFront 프라이빗 키 재포맷(.NET 및 Java만 해당) (p. 121) 단원을 참조하십시오.
2. `Set-Cookie` 헤더 세 개를 승인된 최종 사용자에게 보내도록 애플리케이션을 프로그래밍합니다. 각 `Set-Cookie` 헤더는 이름-값 페어를 하나만 포함할 수 있고 CloudFront 서명된 쿠키에는 이름-값 페어 세 개가 필요하기 때문에 `Set-Cookie` 헤더가 세 개 있어야 합니다. 이름-값 페어는 `CloudFront-`

`Policy`, `CloudFront-Signature`, `CloudFront-Key-Pair-Id`입니다. 액세스가 제한되는 파일에 대해 사용자가 첫 번째 요청을 하려면 최종 사용자에게 값이 있어야 합니다.

> **Note**
>
> 일반적으로 `Expires` 및 `Max-Age` 속성을 제외하는 것이 좋습니다. 이렇게 하면 사용자가 브라우저를 닫을 때 쿠키가 삭제되므로 권한 없는 사람이 콘텐츠에 액세스할 가능성이 줄어듭니다. 자세한 내용은 서명된 쿠키 악용 방지 (p. 143) 단원을 참조하십시오.

쿠키 속성의 이름은 대소문자를 구분합니다.

줄바꿈은 속성의 가독성을 높이기 위해서만 사용됩니다.

```
Set-Cookie:
CloudFront-Policy=base64 encoded version of the policy statement;
Domain=optional domain name;
Path=/optional directory path;
Secure;
HttpOnly


Set-Cookie:
CloudFront-Signature=hashed and signed version of the policy statement;
Domain=optional domain name;
Path=/optional directory path;
Secure;
HttpOnly

Set-Cookie:
CloudFront-Key-Pair-Id=active CloudFront key pair Id for the key pair that you are
 using to generate the signature;
Domain=optional domain name;
Path=/optional directory path;
Secure;
HttpOnly
```

(선택) **`Domain`**

요청된 파일의 도메인 이름입니다. `Domain` 속성을 지정하지 않는 경우, 기본값은 URL의 도메인 이름이고 이는 하위 도메인이 아닌 지정된 도메인 이름에만 적용됩니다. `Domain` 속성을 지정하는 경우, 하위 도메인에도 적용됩니다. 도메인 이름 앞의 점(예: `Domain=.example.com`)은 선택 사항입니다. 또한 `Domain` 속성을 지정하는 경우, URL의 도메인 이름과 `Domain` 속성의 값이 일치해야 합니다.

CloudFront가 배포에 할당하는 도메인 이름을 d111111abcdef8.cloudfront.net과 같이 지정할 수 있으나, *.cloudfront.net을 도메인 이름으로 지정할 수는 없습니다.

URL에 example.com과 같은 대체 도메인 이름을 사용하려면 `Domain` 속성 지정 여부와 상관없이 대체 도메인 이름을 배포에 추가해야 합니다. 자세한 내용은 Alternate Domain Names (CNAMEs) (p. 51) 주제에서 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.

(선택) **`Path`**

요청된 파일의 경로입니다. `Path` 속성을 지정하지 않는 경우, 기본값은 URL의 경로입니다.

`Secure`

요청을 보내기 전 최종 사용자에게 쿠키의 암호화를 요청하세요. 쿠키 속성이 MITM(중간자 공격)을 당하지 않도록 HTTPS 연결을 통해 `Set-Cookie` 헤더를 보내는 것이 좋습니다.

HttpOnly

HTTP 또는 HTTPS 요청에 한해 최종 사용자에게 쿠키를 요구하세요.

CloudFront-Policy

공백이 제거된 JSON 형식의 정책 설명은 base64로 인코딩합니다. 자세한 내용은 사용자 지정 정책을 사용하는 서명된 쿠키에 대한 서명 생성 (p. 154) 단원을 참조하십시오.

정책 설명을 통해 서명된 쿠키가 사용자에게 부여하는 액세스 권한을 제어할 수 있습니다. 이로써 사용자가 액세스할 수 있는 파일, 만료 날짜 및 시간, URL의 효력이 발생하는 선택 날짜 및 시간, 파일에 액세스할 수 있는 선택적 IP 주소 또는 IP 주소 범위 등을 제어합니다.

CloudFront-Signature

JSON 정책 설명을 해시, 서명 및 base64로 인코딩한 버전입니다. 자세한 내용은 사용자 지정 정책을 사용하는 서명된 쿠키에 대한 서명 생성 (p. 154) 단원을 참조하십시오.

CloudFront-Key-Pair-Id

APKA9ONS7QCOWEXAMPLE와 같은 활성 CloudFront 키 페어의 ID입니다. CloudFront는 서명된 쿠키를 확인할 때 사용할 퍼블릭 키를 CloudFront 키 페어 ID로 판단합니다. CloudFront는 서명의 정보를 정책 설명의 정보와 비교하고 URL이 변조되지 않았음을 확인합니다.

CloudFront 서명된 쿠키에 포함하는 키 페어 ID는 캐시 동작에 대한 신뢰할 수 있는 서명자 중 하나인 AWS 계정과 연결되어야 합니다.

자세한 내용은 서명된 URL 및 서명된 쿠키(신뢰할 수 있는 서명자)를 만들 수 있는 AWS 계정 지정 (p. 118) 단원을 참조하십시오.

CloudFront 키 페어를 교체하는 동안 이를 비활성화하고 애플리케이션을 업데이트하여 신뢰할 수 있는 서명자 중 하나에 해당하는 새로운 활성 키 페어를 사용해야 합니다. 키 페어 교체에 대한 자세한 내용은 CloudFront 키 페어 교체 (p. 123)를 참조하십시오.

배포와 연결된 도메인 이름을 파일의 URL에 사용할 때 서명된 쿠키 한 개의 예제 `Set-Cookie` 헤더:

```
Set-Cookie: CloudFront-
Policy=eyJTdGF0ZW1lbnQiOlt7IlJlc291cmNlIjoiaHR0cDovL2QxMTExMTFhYmNkZWY4LmNsb3VkZnJvbnQubmV0L2dhbWVfZG93
 Domain=d111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Signature=dtKhpJ3aUYxqDIwepczPiDb9NXQ_;
 Domain=d111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Key-Pair-Id=APKA9ONS7QCOWEXAMPLE;
 Domain=d111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly
```

파일에 대한 URL의 example.org 대체 도메인 이름을 사용할 때 하나의 서명된 쿠키에 대한 예제 `Set-Cookie` 헤더:

```
Set-Cookie: CloudFront-
Policy=eyJTdGF0ZW1lbnQiOlt7IlJlc291cmNlIjoiaHR0cDovL2QxMTExMTFhYmNkZWY4LmNsb3VkZnJvbnQubmV0L2dhbWVfZG93
 Domain=example.org; Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Signature=dtKhpJ3aUYxqDIwepczPiDb9NXQ_; Domain=example.org; Path=/;
 Secure; HttpOnly
Set-Cookie: CloudFront-Key-Pair-Id=APKA9ONS7QCOWEXAMPLE; Domain=example.org; Path=/;
 Secure; HttpOnly
```

URL에 example.com과 같은 대체 도메인 이름을 사용하려면 `Domain` 속성 지정 여부와 상관없이 대체 도메인 이름을 배포에 추가해야 합니다. 자세한 내용은 Alternate Domain Names (CNAMEs) (p. 51) 주제에서 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.

사용자 지정 정책을 사용하는 서명된 쿠키에 대한 정책 설명 생성

사용자 지정 정책에 대한 정책 설명을 만들려면 다음 절차를 수행하십시오. 다양한 방법으로 파일에 대한 액세스를 제어하는 몇 가지 예제 정책 설명에 대한 내용은 사용자 지정 정책을 사용하는 서명된 쿠키에 대한 예제 정책 설명 (p. 153) 단원을 참조하십시오.

사용자 지정 정책을 사용하는 서명된 쿠키에 대한 정책 설명을 만들려면

1. 다음 JSON 형식을 사용하여 정책 설명을 구성하세요.

```
{
   "Statement": [
      {
         "Resource":"URL of the file",
         "Condition":{
            "DateLessThan":{"AWS:EpochTime":required ending date and time in Unix time
 format and UTC},
            "DateGreaterThan":{"AWS:EpochTime":optional beginning date and time in Unix
 time format and UTC},
            "IpAddress":{"AWS:SourceIp":"optional IP address"}
         }
      }
   ]
}
```

   다음을 참조하십시오.

   - 단 한 개의 명령문만 포함시킬 수 있습니다.
   - UTF-8 문자 인코딩을 사용하세요.
   - 지정된 대로 정확하게 모든 문장 부호 및 파라미터 이름을 포함하세요. 파라미터 이름의 약어는 허용되지 않습니다.
   - `Condition` 섹션의 파라미터 순서는 중요하지 않습니다.
   - `Resource`, `DateLessThan`, `DateGreaterThan`, `IpAddress`에 대한 값 관련 내용은 서명된 쿠키에서 사용자 지정 정책을 위한 정책 설명에서 지정한 값 (p. 152)을 참조하십시오.
2. 정책 설명에서 모든 공백(탭과 줄바꿈 문자 포함)을 제거합니다. 애플리케이션 코드의 문자열에 이스케이프 문자를 포함해야 할 수도 있습니다.
3. MIME base64 인코딩 기준으로 정책 설명을 Base64로 인코딩합니다. 자세한 내용은 RFC 2045, MIME(Multipurpose Internet Mail Extensions) Part One: Format of Internet Message Bodies의 Section 6.8, Base64 Content-Transfer-Encoding을 참조하십시오.
4. URL 쿼리 문자열에서 사용할 수 없는 문자를 유효한 문자로 교체합니다. 아래 표에 사용할 수 없는 문자와 유효한 문자가 나열되어 있습니다.

무효 문자를	교체할 유효한 문자
+	- (하이픈)
=	_ (밑줄)
/	~ (물결표)

5. `Set-Cookie` 뒤 `CloudFront-Policy=` 헤더의 결과 값을 포함하세요.
6. 정책 설명을 해시, 서명 및 base64로 인코딩하여 `Set-Cookie`의 `CloudFront-Signature` 헤더에 대한 서명을 만드세요. 자세한 내용은 사용자 지정 정책을 사용하는 서명된 쿠키에 대한 서명 생성 (p. 154) 단원을 참조하십시오.

서명된 쿠키에서 사용자 지정 정책을 위한 정책 설명에서 지정한 값

사용자 지정 정책에 대한 정책 설명을 생성할 때 다음 값을 지정합니다.

리소스

쿼리 문자열을 포함하는 기본 URL(있는 경우):

```
http://d111111abcdef8.cloudfront.net/images/horizon.jpg?
size=large&license=yes
```

> **Important**
>
> `Resource` 파라미터를 생략하는 경우, 사용자는 서명된 URL을 만들 때 사용하는 키 페어에 연결된 배포 관련 모든 파일에 액세스할 수 있습니다.

`Resource`에 대해 값을 하나만 지정할 수 있습니다.

다음을 참조하십시오.

- 프로토콜 – 이 값은 `http://` 또는 `https://`로 시작해야 합니다.
- 쿼리 문자열 파라미터 – 쿼리 문자열 파라미터가 없는 경우, 물음표를 생략합니다.
- 와일드카드 – 0개 이상의 문자(*)에 해당하거나 문자열 내 정확히 1문자(?)에 해당하는 와일드카드를 문자열 내 모든 곳에 사용할 수 있습니다. 예를 들어 값은

  ```
  http://d111111abcdef8.cloudfront.net/*game_download.zip*
  ```

  다음 파일을 포함합니다.

  - `http://d111111abcdef8.cloudfront.net/game_download.zip`
  - `http://d111111abcdef8.cloudfront.net/example_game_download.zip?license=yes`
  - `http://d111111abcdef8.cloudfront.net/test_game_download.zip?license=temp`
- 대체 도메인 이름 – URL에 대체 도메인 이름(CNAME)을 지정하는 경우, 웹 페이지 또는 애플리케이션의 파일을 참조할 때 대체 도메인 이름을 지정해야 합니다. 파일에 대한 Amazon S3 URL을 지정하지 마십시오.

DateLessThan

Unix 시간 형식(초) 및 협정 세계시(UTC) 기준의 URL 만료 날짜 및 시간. 값을 인용 부호로 묶지 마세요.

예를 들면 2015년 3월 16일 오전 10시(UTC)를 Unix 시간 형식인 1426500000으로 변환합니다.

자세한 내용은 CloudFront는 언제 서명된 쿠키의 만료 날짜 및 시간을 확인합니까? (p. 143) 단원을 참조하십시오.

DateGreaterThan(선택 사항)

Unix 시간 형식(초)과 협정 세계시(UTC)의 URL에 대한 선택적 시작 날짜 및 시간. 지정된 날짜 및 시간 전에 사용자는 파일에 액세스할 수 없습니다. 값을 인용 부호로 묶지 마세요.

IpAddress(선택 사항)

클라이언트의 IP 주소는 GET 요청을 합니다. 다음을 참조하십시오.

- 파일에 액세스하는 IP 주소를 허용하려면 `IpAddress` 파라미터를 생략합니다.
- 하나의 IP 주소 또는 하나의 IP 주소 범위를 지정할 수 있습니다. 예를 들면, 클라이언트의 IP 주소가 두 개의 다른 범위 중 하나에 있는 경우, 정책을 설정하여 액세스를 허용할 수 없습니다.
- 단일 IP 주소에서 액세스를 허용하기 위해 다음을 지정합니다.

  ```
  "IPv4 IP ##/32"
  ```

- IP 주소 범위는 스탠다드 IPv4 CIDR 형식(예: `192.0.2.0/24`)을 지정해야 합니다. 자세한 내용은 RFC 4632, Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan, http://tools.ietf.org/html/rfc4632을 참조하십시오.

  Important

  IPv6 형식의 IP 주소(예: 2001:0db8:85a3:0000:0000:8a2e:0370:7334)는 지원되지 않습니다.

  `IpAddress`를 포함하는 사용자 지정 정책을 사용할 경우 배포에 대해 IPv6를 사용하도록 설정하지 마세요. IP 주소로 일부 콘텐츠에 대한 액세스를 제한하고, 다른 콘텐츠에 대해서는 IPv6 요청을 지원하려면 두 가지 배포를 만들면 됩니다. 자세한 내용은 IPv6 사용 (p. 55) 주제에서 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.

사용자 지정 정책을 사용하는 서명된 쿠키에 대한 예제 정책 설명

다음 예제 정책 설명은 특정 파일, 디렉터리의 모든 파일 또는 키 페어 ID에 연결된 모든 파일의 액세스 제어 방법을 보여줍니다. 예제는 또한 개별 IP 주소 또는 IP 주소 범위에서 액세스를 제어하는 방법과 지정된 날짜 및 시간 이후 사용자의 서명된 쿠키 사용 방지 방법을 보여줍니다.

이러한 예제를 복사하여 붙여 넣는 경우, 공백(탭과 줄바꿈 문자 포함)을 제거하고 값을 자체 값과 교체한 후 닫는 괄호(}) 뒤에 줄바꿈 문자를 포함합니다.

자세한 내용은 서명된 쿠키에서 사용자 지정 정책을 위한 정책 설명에서 지정한 값 (p. 152) 단원을 참조하십시오.

주제

- 예제 정책 설명: IP 주소 범위에서 하나의 파일에 액세스 (p. 153)
- 예제 정책 설명: IP 주소 범위에서 디렉터리의 모든 파일 액세스 (p. 153)
- 예제 정책 설명: 하나의 IP 주소에서 키 페어 ID에 연결된 모든 파일에 액세스 (p. 154)

예제 정책 설명: IP 주소 범위에서 하나의 파일에 액세스

다음 서명된 쿠키의 예제 사용자 지정 정책은 UTC 기준 2013년 1월 1일 오전 10시까지 `192.0.2.0/24` 범위의 IP 주소에서 `http://d111111abcdef8.cloudfront.net/game_download.zip` 파일에 액세스할 수 있는 사용자를 지정합니다.

```
{
   "Statement": [
      {
         "Resource":"http://d111111abcdef8.cloudfront.net/game_download.zip",
         "Condition":{
            "IpAddress":{"AWS:SourceIp":"192.0.2.0/24"},
            "DateLessThan":{"AWS:EpochTime":1357034400}
         }
      }
   ]
}
```

예제 정책 설명: IP 주소 범위에서 디렉터리의 모든 파일 액세스

다음 예제 사용자 지정 정책은 `Resource` 파라미터의 * 와일드카드 문자에 표시된 바와 같이 `training` 디렉터리의 파일에 대한 서명된 쿠키를 만들 수 있도록 허용합니다. 사용자는 UTC 기준 2013년 1월 1일 오전 10시까지 `192.0.2.0/24` 범위의 IP 주소에서 파일에 액세스할 수 있습니다.

```
{
   "Statement": [
```

```
        {
            "Resource":"http://d111111abcdef8.cloudfront.net/training/*",
            "Condition":{
                "IpAddress":{"AWS:SourceIp":"192.0.2.0/24"},
                "DateLessThan":{"AWS:EpochTime":1357034400}
            }
        }
    ]
}
```

이 정책을 사용하는 서명된 쿠키는 각자 특정 파일을 식별하는 다음과 같은 기본 URL을 포함합니다.

`http://d111111abcdef8.cloudfront.net/training/orientation.pdf`

예제 정책 설명: 하나의 IP 주소에서 키 페어 ID에 연결된 모든 파일에 액세스

다음 샘플 사용자 지정 정책은 `Resource` 파라미터의 * 와일드카드 문자 표시된 바와 같이 배포와 연결된 파일에 대한 서명된 쿠키를 설정할 수 있도록 허용합니다. 사용자는 IP 주소 `192.0.2.10/32`를 사용해야 합니다. `192.0.2.10/32` CIDR 표기법의 값은 단일 IP 주소 `192.0.2.10`를 가리킵니다. 파일은 UTC 기준 2013년 1월 1일 오전 10시에서 2013년 1월 2일 오전 10시까지만 사용할 수 있습니다.

```
{
    "Statement": [
        {
            "Resource":"http://*",
            "Condition":{
                "IpAddress":{"AWS:SourceIp":"192.0.2.10/32"},
                "DateGreaterThan":{"AWS:EpochTime":1357034400},
                "DateLessThan":{"AWS:EpochTime":1357120800}
            }
        }
    ]
}
```

이 정책을 사용하는 서명된 쿠키는 각자 특정 CloudFront 배포의 특정 파일을 식별하는 다음과 같은 기본 URL을 포함합니다.

`http://d111111abcdef8.cloudfront.net/training/orientation.pdf`

또한 서명된 쿠키는 키 페어 ID를 포함하고 이는 기본 URL에서 지정한 배포(d111111abcdef8.cloudfront.net)의 신뢰할 수 있는 서명자와 연결되어야 합니다.

사용자 지정 정책을 사용하는 서명된 쿠키에 대한 서명 생성

서명된 쿠키에 대한 서명은 정책 설명의 해시, 서명 및 base64로 인코딩된 버전의 사용자 지정 정책을 사용합니다.

다음을 참조하여 정책 설명을 해시, 서명 및 인코딩하는 방법에 대한 추가적인 내용과 예제를 확인하세요.

- Base64 인코딩 및 암호화를 위한 Linux 명령 및 OpenSSL 사용 (p. 155)
- 서명 URL에 대한 서명을 만드는 코드 예제 (p. 156)
- 비공개 콘텐츠 구성을 위한 도구 및 코드 예제 (p. 454)

사용자 지정 정책으로 서명된 쿠키에 대한 서명을 만들려면

1. SHA-1 해시 함수 및 RSA를 사용하여 사용자 지정 정책을 사용하는 서명된 URL에 대한 정책 설명을 만들려면 (p. 136) 절차에서 만든 JSON 정책 설명을 해시하고 서명합니다. 더 이상 공백을 포함하지 않지만 base64로 인코딩되지 않은 정책 설명 버전을 사용하세요.

해시 함수에 필요한 프라이빗 키의 경우, 신뢰할 수 있는 활성 서명자와 연결된 프라이빗 키를 사용하십시오.

Note

정책 설명을 해시 및 서명하는 방법은 프로그래밍 언어와 플랫폼에 따라 달라집니다. 샘플 코드에 대한 내용은 서명 URL에 대한 서명을 만드는 코드 예제 (p. 156)를 참조하십시오.

2. 해시 및 서명된 문자열에서 모든 공백(탭과 줄바꿈 문자 포함)을 제거합니다.
3. MIME base64 인코딩 기준으로 문자열을 base64로 인코딩합니다. 자세한 내용은 RFC 2045, MIME(Multipurpose Internet Mail Extensions) Part One: Format of Internet Message Bodies의 Section 6.8, Base64 Content-Transfer-Encoding을 참조하십시오.
4. URL 쿼리 문자열에서 사용할 수 없는 문자를 유효한 문자로 교체합니다. 아래 표에 사용할 수 없는 문자와 유효한 문자가 나열되어 있습니다.

무효 문자를	교체할 유효한 문자
+	- (하이픈)
=	_ (밑줄)
/	~ (물결표)

5. `Set-Cookie` 이름-값 페어에 대한 `CloudFront-Signature=` 헤더의 결과 값을 포함하고 사용자 지정 정책을 사용하여 서명된 쿠키를 설정하려면 (p. 148)으로 돌아가서 `Set-Cookie`에 대한 `CloudFront-Key-Pair-Id` 헤더를 추가하세요.

Base64 인코딩 및 암호화를 위한 Linux 명령 및 OpenSSL 사용

다음 Linux 명령줄 명령과 OpenSSL을 사용하여 정책 설명을 해시 및 서명한 후, 서명을 base64로 인코딩하여 URL 쿼리 문자열 파라미터에서 사용할 수 없는 문자를 유효한 문자와 교체하세요.

OpenSSL에 대한 자세한 내용은 http://www.openssl.org를 참조하십시오.

```
(1) cat policy | (2) tr -d "\n" | tr -d " \t\n\r" | (3) openssl sha1 -sign
private-key.pem | (4) openssl base64 | (5) tr -- '+=/' '-_~'
```

여기서 각 항목은 다음과 같습니다.

(1) `cat`은 `policy` 파일을 읽습니다.

(2) `tr -d "\n" | tr -d " \t\n\r"`은(는) `cat`에 의해 추가된 공백과 줄 바꿈 문자를 제거합니다.

(3) OpenSSL은 SHA-1 함수를 사용하여 파일을 해시하고 RSA 및 프라이빗 키 파일 `private-key.pem`을 사용하여 서명합니다.

(4) OpenSSL은 해시, 서명된 정책 설명을 base64로 인코딩합니다.

(5) `tr`은 URL 쿼리 문자열 파라미터에 사용할 수 없는 문자를 유효한 문자로 교체합니다.

몇 가지 프로그래밍 언어로 서명을 만드는 방법을 보여주는 코드 예제는 서명 URL에 대한 서명을 만드는 코드 예제 (p. 156)를 참조하십시오.

서명 URL에 대한 서명을 만드는 코드 예제

이 단원에는 서명된 URL에 대한 서명을 만드는 방법을 담은 애플리케이션 예제가 수록되어 있으며 이를 다운로드할 수 있습니다. 예제는 Perl, PHP, C#, Java로 제공됩니다. 이러한 예제를 사용하여 서명된 URL을 만들 수 있습니다. Perl 스크립트는 Linux/Mac 플랫폼에서 실행됩니다. PHP 예제는 PHP를 실행하는 모든 서버에서 작동합니다. C# 예제는 NET Framework를 사용합니다.

Ruby on Rails로 쿠키를 사용하는 방법에 대한 예제는 비공개 콘텐츠 구성을 위한 도구 및 코드 예제 (p. 454) 주제의 Amazon CloudFront 관련 정보 (p. 453)를 참조하십시오.

다양한 프로그래밍 언어에서 서명된 URL 및 서명된 쿠키에 대한 예제 코드를 찾을 수 있습니다. `sample app ## cloudfront signed URLs` 또는 `sample app ## cloudfront signed cookies`에서 인터넷 검색을 수행하십시오.

주제

- Perl을 사용한 URL 서명 생성 (p. 156)
- PHP를 사용한 URL 서명 생성 (p. 164)
- C# 및 .NET Framework를 사용한 URL 서명 생성 (p. 169)
- Java를 사용한 URL 서명 생성 (p. 174)

Perl을 사용한 URL 서명 생성

이 단원에는 프라이빗 콘텐츠에 대한 서명을 생성하는 데 사용할 수 있는 Linux/Mac 플랫폼용 Perl 스크립트가 포함되어 있습니다. 서명을 생성하려면, CloudFront URL, 서명자의 프라이빗 키 경로, 키 ID와 URL에 대한 만료 날짜를 지정하는 명령줄 인수를 포함한 스크립트를 실행합니다. 또한 이 도구는 서명된 URL을 디코딩할 수 있습니다.

> **Note**
>
> URL 서명 생성은 서명된 URL을 통해 프라이빗 콘텐츠를 제공하는 프로세스의 한 부분에 불과합니다. 종단 간 프로세스에 대한 자세한 내용은 서명된 URL 사용 (p. 124) 단원을 참조하십시오.

주제

- Perl 스크립트를 사용하여 서명된 URL을 생성하는 방법에 대한 예제 (p. 156)
- 서명된 URL을 생성하기 위한 Perl 스크립트의 소스 (p. 157)

Perl 스크립트를 사용하여 서명된 URL을 생성하는 방법에 대한 예제

다음 예제는 이 주제에서 제공하는 Perl 스크립트를 사용하여 RTMP 배포 서명을 만드는 방법을 보여줍니다. 시작하려면, 스크립트를 cfsign.pl이라는 파일로 저장하세요. 그런 다음 명령줄 인수를 사용하여 스크립트를 실행하세요.

```
$ cfsign.pl --action encode --stream example/video.mp4 --private-key
   /path/to/my-private-key.pem --key-pair-id PK12345EXAMPLE --expires 1265838202
```

이 스크립트는 명령줄 인수에서 정책 설명을 만듭니다. 만들어진 서명은 정책 설명의 SHA1 해시입니다.

다음은 base64로 인코딩된 스트림 이름의 예제입니다.

```
mp4:example/video.mp4%3FPolicy%3DewogICJTdGF0ZW1lbnQiOlt7CiAgICAgICJSZXNvdXJjZSI
6ImRyciIsCiAgICAgICJDb25kaXRpb24iOnsKICAgICAgICAiSXBBZGRyZXNzIjp7IkFXUzpTb3VyY2V
JcCI6IjAuMC4wLjAvMCJ9LAogICAgICAgICJEYXRlTGVzc1RoYW4iOnsiQVdTOkVwb2NoVGltZSI6MjE
0NTkxNjgwMH0KICAgICAgfQogICAgEXAMPLE_%26Signature%3DewtHqEXK~68tsZt-eOFnZKGwTf2a
JlbKhXkK5SSiVqcG9pieCRV3xTEPtc29OzeXlsDvRycOM2WK0cXzcyYZhpl9tv2796ihHiCTAwIHQ8yP
17Af4nWtOLIZHoH6wkR3tU1cQHs8R1d-g-SlZGjNBXr~J2MbaJzm8i6EXAMPLE_%26Key-Pair-Id%3
```

```
DPK12345EXAMPLE
```

이 서명은 스트림 프라이빗 콘텐츠 `example/video.mp4`로의 요청을 인증합니다.

Adobe Flash Player를 사용하고 JavaScript를 사용하는 웹 페이지에서 스트림 이름을 전달받은 경우, 서명을 base64로 인코딩하고 URL 요청 파라미터에서 사용하지 못하는 문자(+, =, /)를 유효한 문자(각각 -, _, ~)와 교체합니다.

스트림 이름을 웹 페이지에서 전달받지 못한 경우, 서명을 base64로 인코딩할 필요가 없습니다. 예를 들면 자체 플레이어를 개발하고 스트림 이름을 Adobe Flash .swf 파일 내에서 가져오는 경우, 서명을 base64로 인코딩할 수 없습니다.

다음 예제는 CloudFront를 통해 jwplayer를 사용합니다.

```
<script type='text/javascript'>
  var so1 = new SWFObject
    ('http://d84l721fxaaqy9.cloudfront.net/player/player.swf',
    'mpl', '640', '360', '9');
  so1.addParam('allowfullscreen','true');
  so1.addParam('allowscriptaccess','always');
  so1.addParam('wmode','opaque');
  so1.addVariable('streamer','rtmp://s33r3xe4ayhhis.cloudfront.net/cfx/st');
  so1.addVariable("file","mp4:example/video.mp4%3FPolicy%3DewogICJTdGF0ZW1lbnQi
    Olt7CiAgICAgICJSZXNvdXJjZSI6ImRyciIsCiAgICAgICJDb25kaXRpb24iOnsKICAgICAgICA
    iSXBBZGRyZXNzIjp7IkFXUzpTb3VyY2VJcCI6IjAuMC4wLjAvMCJ9LAogICAgICAgICJEYXRlTG
    Vzc1RoYW4iOnsiQVdTOkVwb2NoVGltZSI6MjE0NTkxNjgwMH0KICAgICAgfQogICAgEXAMPLE_%
    26Signature%3DewtHqEXK~68tsZt-eOFnZKGwTf2aJlbKhXkK5SSiVqcG9pieCRV3xTEPtc29O
    zeXlsDvRycOM2WK0cXzcyYZhpl9tv2796ihHiCTAwIHQ8yP17Af4nWtOLIZHoH6wkR3tU1cQHs8
    R1d-g-SlZGjNBXr~J2MbaJzm8i6EXAMPLE_%26Key-Pair-Id%3DPK12345EXAMPLE
  so1.write('flv');
</script>
```

Adobe Flash .swf 파일에서 스트림을 가져와서 재생하는 경우, 스트림 이름을 URL로 인코딩하지 마세요. 예:

```
mp4:example/video.mp4?Policy=ewogICJTdGF0ZW1lbnQiOlt7CiAgICAgICJSZXNvdXJjZSI6ImR
yciIsCiAgICAgICJDb25kaXRpb24iOnsKICAgICAgICAiSXBBZGRyZXNzIjp7IkFXUzpTb3VyY2VJcCI
6IjAuMC4wLjAvMCJ9LAogICAgICAgICJEYXRlTGVzc1RoYW4iOnsiQVdTOkVwb2NoVGltZSI6MjE0NTk
xNjgwMH0KICAgICAgfQogICAgEXAMPLE_&Signature=ewtHqEXK~68tsZt-eOFnZKGwTf2aJlbKhXkK
5SSiVqcG9pieCRV3xTEPtc29OzeXlsDvRycOM2WK0cXzcyYZhpl9tv2796ihHiCTAwIHQ8yP17Af4nWt
OLIZHoH6wkR3tU1cQHs8R1d-g-SlZGjNBXr~J2MbaJzm8i6EXAMPLE_&Key-Pair-Id=PK12345
EXAMPLE
```

명령줄 전환 및 이 도구의 기능에 대한 자세한 내용은 다음 단원에 포함되어 있는 Perl 소스 코드의 명령을 참조하십시오.

다음 사항도 참조하세요.

- PHP를 사용한 URL 서명 생성 (p. 164)
- C# 및 .NET Framework를 사용한 URL 서명 생성 (p. 169)
- Java를 사용한 URL 서명 생성 (p. 174)
- 비공개 콘텐츠 구성을 위한 도구 및 코드 예제 (p. 454)

서명된 URL을 생성하기 위한 Perl 스크립트의 소스

서명된 CloudFront URL을 만들기 위해 다음과 같은 Perl 소스 코드를 사용할 수 있습니다. 이 코드의 명령에는 명령줄 전환 및 이 도구의 기능에 대한 자세한 내용이 포함되어 있습니다.

```
#!/usr/bin/perl -w
```

```
# Copyright 2008 Amazon Technologies, Inc.  Licensed under the Apache License, Version 2.0
 (the "License");
# you may not use this file except in compliance with the License. You may obtain a copy of
 the License at:
#
# http://aws.amazon.com/apache2.0
#
# This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
 KIND, either express or implied.
# See the License for the specific language governing permissions and limitations under the
 License.

=head1 cfsign.pl

cfsign.pl - A tool to generate and verify AWS CloudFront signed URLs

=head1 SYNOPSIS

This script uses an existing RSA key pair to sign and verify AWS CloudFront signed URLs

View the script source for details as to which CPAN packages are required beforehand.

For help, try:

cfsign.pl --help

URL signing examples:

cfsign.pl --action encode --url http://images.my-website.com/gallery1.zip --policy
 sample_policy.json --private-key privkey.pem --key-pair-id mykey

cfsign.pl --action encode --url http://images.my-website.com/gallery1.zip --expires
 1257439868 --private-key privkey.pem --key-pair-id mykey

Stream signing example:

cfsign.pl --action encode --stream videos/myvideo.mp4 --expires 1257439868 --private-key
 privkey.pem --key-pair-id mykey

URL decode example:

cfsign.pl --action decode --url "http//mydist.cloudfront.net/?Signature=AGO-
PgxkYo99MkJFHvjfGXjG1QDEXeaDb4Qtzmy85wqyJjK7eKojQWa4BCRcow__&Policy=eyJTdGF0ZW1lbnQiOlt7IlJlc291cmNlIjo
Pair-Id=mykey"


To generate an RSA key pair, you can use openssl and the following commands:

# Generate a 1024bit key pair
openssl genrsa -out private-key.pem 1024
openssl rsa -in private-key.pem -pubout -out public-key.pem


=head1 OPTIONS

=over 8

=item B<--help>

Print a help message and exits.

=item B<--action> [action]

The action to execute.  action can be one of:
```

```
  encode - Generate a signed URL (using a canned policy or a user policy)
  decode - Decode a signed URL

=item B<--url>

The URL to en/decode

=item B<--stream>

The stream to en/decode

=item B<--private-key>

The path to your private key.

=item B<--key-pair-id>

The AWS Portal assigned key pair identifier.

=item B<--policy>

The CloudFront policy document.

=item B<--expires>

The Unix epoch time when the URL is to expire. If both this option and
the --policy option are specified, --policy will be used. Otherwise, this
option alone will use a canned policy.

=back

=cut

use strict;
use warnings;

# you might need to use CPAN to get these modules.
# run perl -MCPAN -e "install <module>" to get them.
# The openssl command line will also need to be in your $PATH.
use File::Temp qw/tempfile/;
use File::Slurp;
use Getopt::Long;
use IPC::Open2;
use MIME::Base64 qw(encode_base64 decode_base64);
use Pod::Usage;
use URI;

my $CANNED_POLICY
    = '{"Statement":[{"Resource":"<RESOURCE>","Condition":{"DateLessThan":
{"AWS:EpochTime":<EXPIRES>}}}]}';

my $POLICY_PARAM      = "Policy";
my $EXPIRES_PARAM     = "Expires";
my $SIGNATURE_PARAM   = "Signature";
my $KEY_PAIR_ID_PARAM = "Key-Pair-Id";

my $verbose = 0;
my $policy_filename = "";
my $expires_epoch = 0;
my $action = "";
my $help = 0;
my $key_pair_id = "";
my $url = "";
my $stream = "";
my $private_key_filename = "";
```

```
my $result = GetOptions("action=s"      => \$action,
                        "policy=s"      => \$policy_filename,
                        "expires=i"     => \$expires_epoch,
                        "private-key=s" => \$private_key_filename,
                        "key-pair-id=s" => \$key_pair_id,
                        "verbose"       => \$verbose,
                        "help"          => \$help,
                        "url=s"         => \$url,
                        "stream=s"      => \$stream,
                    );

if ($help or !$result) {
    pod2usage(1);
    exit;
}

if ($url eq "" and $stream eq "") {
    print STDERR "Must include a stream or a URL to encode or decode with the --stream or
 --url option\n";
    exit;
}

if ($url ne "" and $stream ne "") {
    print STDERR "Only one of --url and --stream may be specified\n";
    exit;
}

if ($url ne "" and !is_url_valid($url)) {
    exit;
}

if ($stream ne "") {
    exit unless is_stream_valid($stream);

    # The signing mechanism is identical, so from here on just pretend we're
    # dealing with a URL
    $url = $stream;
}

if ($action eq "encode") {
    # The encode action will generate a private content URL given a base URL,
    # a policy file (or an expires timestamp) and a key pair id parameter
    my $private_key;
    my $public_key;
    my $public_key_file;

    my $policy;
    if ($policy_filename eq "") {
        if ($expires_epoch == 0) {
            print STDERR "Must include policy filename with --policy argument or an
 expires" .
                         "time using --expires\n";
        }

        $policy = $CANNED_POLICY;
        $policy =~ s/<EXPIRES>/$expires_epoch/g;
        $policy =~ s/<RESOURCE>/$url/g;
    } else {
        if (! -e $policy_filename) {
            print STDERR "Policy file $policy_filename does not exist\n";
            exit;
        }
        $expires_epoch = 0; # ignore if set
        $policy = read_file($policy_filename);
    }
```

```
    if ($private_key_filename eq "") {
        print STDERR "You must specific the path to your private key file with --private-
key\n";
        exit;
    }

    if (! -e $private_key_filename) {
        print STDERR "Private key file $private_key_filename does not exist\n";
        exit;
    }

    if ($key_pair_id eq "") {
        print STDERR "You must specify an AWS portal key pair id with --key-pair-id\n";
        exit;
    }

    my $encoded_policy = url_safe_base64_encode($policy);
    my $signature = rsa_sha1_sign($policy, $private_key_filename);
    my $encoded_signature = url_safe_base64_encode($signature);

    my $generated_url = create_url($url, $encoded_policy, $encoded_signature, $key_pair_id,
 $expires_epoch);


    if ($stream ne "") {
        print "Encoded stream (for use within a swf):\n" . $generated_url . "\n";
        print "Encoded and escaped stream (for use on a webpage):\n" .
 escape_url_for_webpage($generated_url) . "\n";
    } else {
        print "Encoded URL:\n" . $generated_url . "\n";
    }
} elsif ($action eq "decode") {
    my $decoded = decode_url($url);
    if (!$decoded) {
        print STDERR "Improperly formed URL\n";
        exit;
    }

    print_decoded_url($decoded);
} else {
    # No action specified, print help.  But only if this is run as a program (caller will
 be empty)
    pod2usage(1) unless caller();
}

# Decode a private content URL into its component parts
sub decode_url {
    my $url = shift;

    if ($url =~ /(.*)\?(.*)/) {
        my $base_url = $1;
        my $params = $2;

        my @unparsed_params = split(/&/, $params);
        my %params = ();
        foreach my $param (@unparsed_params) {
            my ($key, $val) = split(/=/, $param);
            $params{$key} = $val;
        }

        my $encoded_signature = "";
        if (exists $params{$SIGNATURE_PARAM}) {
            $encoded_signature = $params{"Signature"};
        } else {
            print STDERR "Missing Signature URL parameter\n";
            return 0;
```

```
        }

        my $encoded_policy = "";
        if (exists $params{$POLICY_PARAM}) {
            $encoded_policy = $params{$POLICY_PARAM};
        } else {
            if (!exists $params{$EXPIRES_PARAM}) {
                print STDERR "Either the Policy or Expires URL parameter needs to be
 specified\n";
                return 0;
            }

            my $expires = $params{$EXPIRES_PARAM};

            my $policy = $CANNED_POLICY;
            $policy =~ s/<EXPIRES>/$expires/g;

            my $url_without_cf_params = $url;
            $url_without_cf_params =~ s/$SIGNATURE_PARAM=[^&]*&?//g;
            $url_without_cf_params =~ s/$POLICY_PARAM=[^&]*&?//g;
            $url_without_cf_params =~ s/$EXPIRES_PARAM=[^&]*&?//g;
            $url_without_cf_params =~ s/$KEY_PAIR_ID_PARAM=[^&]*&?//g;

            if ($url_without_cf_params =~ /(.*)\?$/) {
                $url_without_cf_params = $1;
            }

            $policy =~ s/<RESOURCE>/$url_without_cf_params/g;

            $encoded_policy = url_safe_base64_encode($policy);
        }

        my $key = "";
        if (exists $params{$KEY_PAIR_ID_PARAM}) {
            $key = $params{$KEY_PAIR_ID_PARAM};
        } else {
            print STDERR "Missing $KEY_PAIR_ID_PARAM parameter\n";
            return 0;
        }

        my $policy = url_safe_base64_decode($encoded_policy);

        my %ret = ();
        $ret{"base_url"} = $base_url;
        $ret{"policy"} = $policy;
        $ret{"key"} = $key;

        return \%ret;
    } else {
        return 0;
    }
}

# Print a decoded URL out
sub print_decoded_url {
    my $decoded = shift;

    print "Base URL: \n" . $decoded->{"base_url"} . "\n";
    print "Policy: \n" . $decoded->{"policy"} . "\n";
    print "Key: \n" . $decoded->{"key"} . "\n";
}

# Encode a string with base 64 encoding and replace some invalid URL characters
sub url_safe_base64_encode {
    my ($value) = @_;
```

```
    my $result = encode_base64($value);
    $result =~ tr|+=/|-_~|;

    return $result;
}

# Decode a string with base 64 encoding.  URL-decode the string first
# followed by reversing any special character ("+=/") translation.
sub url_safe_base64_decode {
    my ($value) = @_;

    $value =~ s/%([0-9A-Fa-f]{2})/chr(hex($1))/eg;
    $value =~ tr|-_~|+=/|;

    my $result = decode_base64($value);

    return $result;
}

# Create a private content URL
sub create_url {
    my ($path, $policy, $signature, $key_pair_id, $expires) = @_;

    my $result;
    my $separator = $path =~ /\?/ ? '&' : '?';
    if ($expires) {
        $result = "$path$separator$EXPIRES_PARAM=$expires&$SIGNATURE_PARAM=$signature&
$KEY_PAIR_ID_PARAM=$key_pair_id";
    } else {
        $result = "$path$separator$POLICY_PARAM=$policy&$SIGNATURE_PARAM=$signature&
$KEY_PAIR_ID_PARAM=$key_pair_id";
    }
    $result =~ s/\n//g;

    return $result;
}

# Sign a document with given private key file.
# The first argument is the document to sign
# The second argument is the name of the private key file
sub rsa_sha1_sign {
    my ($to_sign, $pvkFile) = @_;
    print "openssl sha1 -sign $pvkFile $to_sign\n";

    return write_to_program($pvkFile, $to_sign);
}

# Helper function to write data to a program
sub write_to_program {
my ($keyfile, $data) = @_;
unlink "temp_policy.dat" if (-e "temp_policy.dat");
unlink "temp_sign.dat" if (-e "temp_sign.dat");

write_file("temp_policy.dat", $data);

system("openssl dgst -sha1 -sign \"$keyfile\" -out temp_sign.dat temp_policy.dat");

my $output = read_file("temp_sign.dat");

    return $output;
}

# Read a file into a string and return the string
sub read_file {
    my ($file) = @_;
```

```
    open(INFILE, "<$file") or die("Failed to open $file: $!");
    my $str = join('', <INFILE>);
    close INFILE;

    return $str;
}

sub is_url_valid {
    my ($url) = @_;

    # HTTP distributions start with http[s]:// and are the correct thing to sign
    if ($url =~ /^https?:\/\//) {
        return 1;
    } else {
        print STDERR "CloudFront requires absolute URLs for HTTP distributions\n";
        return 0;
    }
}

sub is_stream_valid {
    my ($stream) = @_;

    if ($stream =~ /^rtmp:\/\// or $stream =~ /^\/?cfx\/st/) {
        print STDERR "Streaming distributions require that only the stream name is signed.
\n";
        print STDERR "The stream name is everything after, but not including, cfx/st/\n";
        return 0;
    } else {
        return 1;
    }
}

# flash requires that the query parameters in the stream name are url
# encoded when passed in through javascript, etc.  This sub handles the minimal
# required url encoding.
sub escape_url_for_webpage {
    my ($url) = @_;

    $url =~ s/\?/%3F/g;
    $url =~ s/=/%3D/g;
    $url =~ s/&/%26/g;

    return $url;
}

1;
```

PHP를 사용한 URL 서명 생성

PHP를 실행하는 모든 웹 서버는 이 PHP 예제 코드를 사용하여 프라이빗 CloudFront 배포에 대한 정책 설명 및 서명을 만듭니다. 예제 전체에서는 CloudFront 스트리밍을 사용하여 비디오 스트림을 재생하는 서명된 URL 링크가 있는 정상 웹 페이지를 만듭니다. https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/samples/demo-php.zip에서 예제 전체를 다운로드할 수 있습니다.

PHP용 AWS SDK에서 `UrlSigner` 클래스를 사용하여 서명된 URL을 생성할 수도 있습니다. 자세한 내용은 PHP용 AWS SDK API Reference의 Class UrlSigner를 참조하십시오.

> **Note**
>
> URL 서명 생성은 서명된 URL을 통해 프라이빗 콘텐츠를 제공하는 프로세스의 한 부분에 불과합니다. 전체 프로세스에 대한 자세한 내용은 서명된 URL 사용 (p. 124)을 참조하십시오.

주제

- 샘플: RSA SHA-1 서명 (p. 165)
- 샘플: 미리 준비된 정책 생성 (p. 165)
- 샘플: 사용자 지정 정책 생성 (p. 166)
- 전체 코드 예제 (p. 166)

샘플: RSA SHA-1 서명

다음 코드 샘플에서 함수 `rsa_sha1_sign`는 정책 설명을 해싱 및 서명합니다. 필요한 인수는 배포의 신뢰할 수 있는 서명자인 AWS 계정의 정책 설명과 프라이빗 키입니다. 다음으로 `url_safe_base64_encode` 함수는 서명의 안전한 URL 버전을 만듭니다.

```
function rsa_sha1_sign($policy, $private_key_filename) {
    $signature = "";

    // load the private key
    $fp = fopen($private_key_filename, "r");
    $priv_key = fread($fp, 8192);
    fclose($fp);
    $pkeyid = openssl_get_privatekey($priv_key);

    // compute signature
    openssl_sign($policy, $signature, $pkeyid);

    // free the key from memory
    openssl_free_key($pkeyid);

    return $signature;
}

function url_safe_base64_encode($value) {
    $encoded = base64_encode($value);
    // replace unsafe characters +, = and / with
    // the safe characters -, _ and ~
    return str_replace(
        array('+', '=', '/'),
        array('-', '_', '~'),
        $encoded);
}
```

샘플: 미리 준비된 정책 생성

다음 샘플 코드는 서명에 대한 미리 준비된 정책 설명을 구성합니다. 미리 준비된 정책에 대한 자세한 내용은 미리 준비된 정책을 사용하여 서명된 URL 생성 (p. 127)을 참조하십시오.

> **Note**
>
> `$expires` 변수는 문자열이 아닌 정수여야 하는 날짜/타임 스탬프입니다.

```
function get_canned_policy_stream_name($video_path, $private_key_filename, $key_pair_id,
 $expires) {
    // this policy is well known by CloudFront, but you still need to sign it,
    // since it contains your parameters
    $canned_policy = '{"Statement":[{"Resource":"' . $video_path . '","Condition":
{"DateLessThan":{"AWS:EpochTime":'. $expires . '}}}]}';

    // sign the canned policy
    $signature = rsa_sha1_sign($canned_policy, $private_key_filename);
    // make the signature safe to be included in a url
    $encoded_signature = url_safe_base64_encode($signature);
```

```
    // combine the above into a stream name
    $stream_name = create_stream_name($video_path, null, $encoded_signature, $key_pair_id,
 $expires);
    // url-encode the query string characters to work around a flash player bug
    return encode_query_params($stream_name);
}
```

샘플: 사용자 지정 정책 생성

다음 샘플 코드는 서명에 대한 사용자 지정 정책 설명을 구성합니다. 사용자 지정 정책에 대한 내용은 사용자 지정 정책을 사용한 서명된 URL 생성 (p. 133)을 참조하십시오.

```
function get_custom_policy_stream_name($video_path, $private_key_filename, $key_pair_id,
 $policy) {
    // sign the policy
    $signature = rsa_sha1_sign($policy, $private_key_filename);
    // make the signature safe to be included in a url
    $encoded_signature = url_safe_base64_encode($signature);

    // combine the above into a stream name
    $stream_name = create_stream_name($video_path, $encoded_policy, $encoded_signature,
 $key_pair_id, null);
    // url-encode the query string characters to work around a flash player bug
    return encode_query_params($stream_name);
}
```

전체 코드 예제

다음 예제 코드는 PHP로 CloudFront 서명 URL을 만드는 전체 예제를 제공합니다. https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/samples/demo-php.zip에서 이 전체 예제를 다운로드할 수 있습니다.

```
<?php

function rsa_sha1_sign($policy, $private_key_filename) {
    $signature = "";

    // load the private key
    $fp = fopen($private_key_filename, "r");
    $priv_key = fread($fp, 8192);
    fclose($fp);
    $pkeyid = openssl_get_privatekey($priv_key);

    // compute signature
    openssl_sign($policy, $signature, $pkeyid);

    // free the key from memory
    openssl_free_key($pkeyid);

    return $signature;
}

function url_safe_base64_encode($value) {
    $encoded = base64_encode($value);
    // replace unsafe characters +, = and / with the safe characters -, _ and ~
    return str_replace(
        array('+', '=', '/'),
        array('-', '_', '~'),
        $encoded);
}

function create_stream_name($stream, $policy, $signature, $key_pair_id, $expires) {
```

```
    $result = $stream;
    // if the stream already contains query parameters, attach the new query parameters to
 the end
    // otherwise, add the query parameters
    $separator = strpos($stream, '?') == FALSE ? '?' : '&';
    // the presence of an expires time means we're using a canned policy
    if($expires) {
        $result .= $path . $separator . "Expires=" . $expires . "&Signature=" .
 $signature . "&Key-Pair-Id=" . $key_pair_id;
    }
    // not using a canned policy, include the policy itself in the stream name
    else {
        $result .= $path . $separator . "Policy=" . $policy . "&Signature=" . $signature .
 "&Key-Pair-Id=" . $key_pair_id;
    }

    // new lines would break us, so remove them
    return str_replace('\n', '', $result);
}

function encode_query_params($stream_name) {
    // Adobe Flash Player has trouble with query parameters being passed into it,
    // so replace the bad characters with their URL-encoded forms
    return str_replace(
        array('?', '=', '&'),
        array('%3F', '%3D', '%26'),
        $stream_name);
}

function get_canned_policy_stream_name($video_path, $private_key_filename, $key_pair_id,
 $expires) {
    // this policy is well known by CloudFront, but you still need to sign it, since it
 contains your parameters
    $canned_policy = '{"Statement":[{"Resource":"' . $video_path . '","Condition":
{"DateLessThan":{"AWS:EpochTime":'. $expires . '}}}]}';
    // the policy contains characters that cannot be part of a URL, so we base64 encode it
    $encoded_policy = url_safe_base64_encode($canned_policy);
    // sign the original policy, not the encoded version
    $signature = rsa_sha1_sign($canned_policy, $private_key_filename);
    // make the signature safe to be included in a URL
    $encoded_signature = url_safe_base64_encode($signature);

    // combine the above into a stream name
    $stream_name = create_stream_name($video_path, null, $encoded_signature, $key_pair_id,
 $expires);
    // URL-encode the query string characters to support Flash Player
    return encode_query_params($stream_name);
}

function get_custom_policy_stream_name($video_path, $private_key_filename, $key_pair_id,
 $policy) {
    // the policy contains characters that cannot be part of a URL, so we base64 encode it
    $encoded_policy = url_safe_base64_encode($policy);
    // sign the original policy, not the encoded version
    $signature = rsa_sha1_sign($policy, $private_key_filename);
    // make the signature safe to be included in a URL
    $encoded_signature = url_safe_base64_encode($signature);

    // combine the above into a stream name
    $stream_name = create_stream_name($video_path, $encoded_policy, $encoded_signature,
 $key_pair_id, null);
    // URL-encode the query string characters to support Flash Player
    return encode_query_params($stream_name);
}
```

```
// Path to your private key.  Be very careful that this file is not accessible
// from the web!

$private_key_filename = '/home/test/secure/example-priv-key.pem';
$key_pair_id = 'AKIAIOSFODNN7EXAMPLE';

$video_path = 'example.mp4';

$expires = time() + 300; // 5 min from now
$canned_policy_stream_name = get_canned_policy_stream_name($video_path,
 $private_key_filename, $key_pair_id, $expires);

$client_ip = $_SERVER['REMOTE_ADDR'];
$policy =
'{'.
    '"Statement":['.
        '{'.
            '"Resource":"'. $video_path . '",'.
            '"Condition":{'.
                '"IpAddress":{"AWS:SourceIp":"' . $client_ip . '/32"},'.
                '"DateLessThan":{"AWS:EpochTime":' . $expires . '}'.
            '}'.
        '}'.
    ']' .
    '}';
$custom_policy_stream_name = get_custom_policy_stream_name($video_path,
 $private_key_filename, $key_pair_id, $policy);

?>

<html>

<head>
    <title>CloudFront</title>
<script type='text/javascript' src='https://example.cloudfront.net/player/swfobject.js'></
script>
</head>

<body>
    <h1>Amazon CloudFront</h1>
    <h2>Canned Policy</h2>
    <h3>Expires at <?= gmdate('Y-m-d H:i:s T', $expires) ?></h3>
    <br />

    <div id='canned'>The canned policy video will be here</div>

    <h2>Canned Policy</h2>
    <h3>Expires at <?= gmdate('Y-m-d H:i:s T', $expires) ?> only viewable by IP <?=
 $client_ip ?></h3>
    <div id='custom'>The custom policy video will be here</div>

    <!-- ************** Have to update the player.swf path to a real JWPlayer instance.
    The fake one means that external people cannot watch the video right now -->
    <script type='text/javascript'>
  var so_canned = new SWFObject('https://files.example.com/
player.swf','mpl','640','360','9');
  so_canned.addParam('allowfullscreen','true');
  so_canned.addParam('allowscriptaccess','always');
  so_canned.addParam('wmode','opaque');
  so_canned.addVariable('file','<?= $canned_policy_stream_name ?>');
  so_canned.addVariable('streamer','rtmp://example.cloudfront.net/cfx/st');
        so_canned.write('canned');

  var so_custom = new SWFObject('https://files.example.com/
player.swf','mpl','640','360','9');
  so_custom.addParam('allowfullscreen','true');
```

```
  so_custom.addParam('allowscriptaccess','always');
  so_custom.addParam('wmode','opaque');
  so_custom.addVariable('file','<?= $custom_policy_stream_name ?>');
  so_custom.addVariable('streamer','rtmp://example.cloudfront.net/cfx/st');
  so_custom.write('custom');
    </script>
</body>

</html>
```

추가 참고:

- Perl을 사용한 URL 서명 생성 (p. 156)
- C# 및 .NET Framework를 사용한 URL 서명 생성 (p. 169)
- Java를 사용한 URL 서명 생성 (p. 174)
- 비공개 콘텐츠 구성을 위한 도구 및 코드 예제 (p. 454)

C# 및 .NET Framework를 사용한 URL 서명 생성

이 섹션의 C# 예제는 사용자 지정 및 미리 준비된 정책 설명을 사용하여 CloudFront 프라이빗 배포에 대한 서명을 만드는 방법을 보여주는 예제 애플리케이션을 구현합니다. 예제는 .NET 애플리케이션에서 유용할 수 있는 AWS .NET SDK를 기반으로 한 유틸리티 함수를 포함합니다.

.NET용 AWS SDK를 사용하여 서명된 URL과 서명된 쿠키를 생성할 수도 있습니다. .NET용 AWS SDK API Reference에서 다음 주제를 참조하십시오.

- 서명된 URL – Amazon.CloudFront > AmazonCloudFrontUrlSigner
- 서명된 쿠키 – Amazon.CloudFront > AmazonCloudFrontUrlSigner

> **Note**
>
> URL 서명 생성은 서명된 URL을 통해 프라이빗 콘텐츠를 제공하는 프로세스의 한 부분에 불과합니다. 전체 프로세스에 대한 자세한 내용은 서명된 URL 사용 (p. 124)을 참조하십시오.

코드를 다운로드하려면 C#의 서명 코드를 참조하십시오.

.NET framework의 AWS 계정/보안에서 제공되는 RSA 키를 사용하려면 AWS-supplied .pem 파일을 .NET framework가 사용하는 XML 형식으로 전환해야 합니다.

변환 후, RSA 프라이빗 키 파일의 형식은 다음과 같습니다.

Example XML .NET Framework 형식의 RSA 프라이빗 키

```
<RSAKeyValue>
  <Modulus>
    wO5IvYCP5UcoCKDo1dcspoMehWBZcyfs9QEzGi6Oe5y+ewGr1oW+vB2GPB
    ANBiVPcUHTFWhwaIBd3oglmF0lGQljP/jOfmXHUK2kUUnLnJp+oOBL2NiuFtqcW6h/L5lIpD8Yq+NRHg
    Ty4zDsyr2880MvXv88yEFURCkqEXAMPLE=
  </Modulus>
  <Exponent>AQAB</Exponent>
  <P>
    5bmKDaTz
    npENGVqz4Cea8XPH+sxt+2VaAwYnsarVUoSBeVt8WLloVuZGG9IZYmH5KteXEu7fZveYd9UEXAMPLE==
  </P>
  <Q>
    1v9l/WN1a1N3rOK4VGoCokx7kR2SyTMSbZgF9IWJNOugR/WZw7HTnjipO3c9dy1Ms9pUKwUF4
    6d7049EXAMPLE==
  </Q>
```

```
  <DP>
    RgrSKuLWXMyBH+/l1Dx/I4tXuAJIrlPyo+VmiOc7b5NzHptkSHEPfR9s1
    OK0Vqjknclq CJ3Ig86OMEtEXAMPLE==
  </DP>
  <DQ>
    pjPjvSFw+RoaTu0pgCA/jwW/FGyfN6iim1RFbkT4
    z49DZb2IM885f3vf35eLTaEYRYUHQgZtChNEV0TEXAMPLE==
  </DQ>
  <InverseQ>
    nkvOJTg5QtGNgWb9i
    cVtzrL/1pFEOHbJXwEJdU99N+7sMK+1066DL/HSBUCD63qD4USpnf0myc24in0EXAMPLE==</InverseQ>
  <D>
      Bc7mp7XYHynuPZxChjWNJZIq+A73gm0ASDv6At7F8Vi9r0xUlQe/v0AQS3ycN8QlyR4XMbzMLYk
      3yjxFDXo4ZKQtOGzLGteCU2srANiLv26/imXA8FVidZftTAtLviWQZBVPTeYIA69ATUYPEq0a5u5wjGy
      UOij9OWyuEXAMPLE=
   </D>
</RSAKeyValue>
```

다음 C# 코드는 다음을 수행하여 미리 준비된 정책을 사용하는 서명된 URL을 만듭니다.

- 정책 설명을 만듭니다.
- AWS 계정 또는 지정된 신뢰할 수 있는 AWS 계정에 대한 SHA1 함수를 사용하는 정책 설명을 해시하고 RSA 및 프라이빗 키를 사용하는 결과에 서명하세요.
- Base64 방식으로 해시 및 서명된 정책 설명을 인코딩하고 특수 문자를 교체하여 URL 요청 파라미터로 사용하기에 안전한 문자열을 만드세요.
- 값을 연결합니다.

완전한 구현에 대한 내용은 C#의 서명 코드에서 예제를 참조하십시오.

Example C#의 미리 준비된 정책 서명 메서드

```
public static string ToUrlSafeBase64String(byte[] bytes)
{
    return System.Convert.ToBase64String(bytes)
        .Replace('+', '-')
        .Replace('=', '_')
        .Replace('/', '~');
}

public static string CreateCannedPrivateURL(string urlString,
    string durationUnits, string durationNumber, string pathToPolicyStmnt,
    string pathToPrivateKey, string privateKeyId)
{
    // args[] 0-thisMethod, 1-resourceUrl, 2-seconds-minutes-hours-days
    // to expiration, 3-numberOfPreviousUnits, 4-pathToPolicyStmnt,
    // 5-pathToPrivateKey, 6-PrivateKeyId

    TimeSpan timeSpanInterval = GetDuration(durationUnits, durationNumber);

    // Create the policy statement.
    string strPolicy = CreatePolicyStatement(pathToPolicyStmnt,
        urlString,
        DateTime.Now,
        DateTime.Now.Add(timeSpanInterval),
        "0.0.0.0/0");
    if ("Error!" == strPolicy) return "Invalid time frame." +
        "Start time cannot be greater than end time.";

    // Copy the expiration time defined by policy statement.
    string strExpiration = CopyExpirationTimeFromPolicy(strPolicy);
```

```
    // Read the policy into a byte buffer.
    byte[] bufferPolicy = Encoding.ASCII.GetBytes(strPolicy);

    // Initialize the SHA1CryptoServiceProvider object and hash the policy data.
    using (SHA1CryptoServiceProvider
        cryptoSHA1 = new SHA1CryptoServiceProvider())
    {
        bufferPolicy = cryptoSHA1.ComputeHash(bufferPolicy);

        // Initialize the RSACryptoServiceProvider object.
        RSACryptoServiceProvider providerRSA = new RSACryptoServiceProvider();
        XmlDocument xmlPrivateKey = new XmlDocument();

        // Load PrivateKey.xml, which you created by converting your
        // .pem file to the XML format that the .NET framework uses.
        // Several tools are available.
        xmlPrivateKey.Load(pathToPrivateKey);

        // Format the RSACryptoServiceProvider providerRSA and
        // create the signature.
        providerRSA.FromXmlString(xmlPrivateKey.InnerXml);
        RSAPKCS1SignatureFormatter rsaFormatter =
            new RSAPKCS1SignatureFormatter(providerRSA);
        rsaFormatter.SetHashAlgorithm("SHA1");
        byte[] signedPolicyHash = rsaFormatter.CreateSignature(bufferPolicy);

        // Convert the signed policy to URL-safe base64 encoding and
        // replace unsafe characters + = / with the safe characters - _ ~
        string strSignedPolicy = ToUrlSafeBase64String(signedPolicyHash);

        // Concatenate the URL, the timestamp, the signature,
        // and the key pair ID to form the signed URL.
        return urlString +
            "?Expires=" +
            strExpiration +
            "&Signature=" +
            strSignedPolicy +
            "&Key-Pair-Id=" +
            privateKeyId;
    }
}
```

다음 C# 코드는 다음을 수행하여 사용자 지정 정책을 사용하는 서명된 URL을 만듭니다.

1. 정책 설명을 만듭니다.
2. Base64 방식으로 정책 설명을 인코딩하고 특수 문자를 교체하여 URL 요청 파라미터로 사용하기에 안전한 문자열을 만드세요.
3. AWS 계정 또는 지정된 신뢰할 수 있는 AWS 계정에 대하여 SHA1 함수를 사용하여 정책 설명을 해시하고 RSA 및 프라이빗 키를 사용하는 결과를 암호화하세요.
4. Base64 방식으로 해시된 정책 설명을 인코딩하고 특수 문자를 교체하여 URL 요청 파라미터로 사용하기에 안전한 문자열을 만드세요.
5. 값을 연결합니다.

완전한 구현에 대한 내용은 C#의 서명 코드에서 예제를 참조하십시오.

Example C#의 사용자 지정 정책 서명 메서드

```
public static string ToUrlSafeBase64String(byte[] bytes)
{
```

```
    return System.Convert.ToBase64String(bytes)
        .Replace('+', '-')
        .Replace('=', '_')
        .Replace('/', '~');
}

public static string CreateCustomPrivateURL(string urlString,
    string durationUnits, string durationNumber, string startIntervalFromNow,
    string ipaddress, string pathToPolicyStmnt, string pathToPrivateKey,
    string PrivateKeyId)
{
    // args[] 0-thisMethod, 1-resourceUrl, 2-seconds-minutes-hours-days
    // to expiration, 3-numberOfPreviousUnits, 4-starttimeFromNow,
    // 5-ip_address, 6-pathToPolicyStmt, 7-pathToPrivateKey, 8-privateKeyId

    TimeSpan timeSpanInterval = GetDuration(durationUnits, durationNumber);
    TimeSpan timeSpanToStart = GetDurationByUnits(durationUnits,
        startIntervalFromNow);
    if (null == timeSpanToStart)
        return "Invalid duration units." +
            "Valid options: seconds, minutes, hours, or days";

    string strPolicy = CreatePolicyStatement(
        pathToPolicyStmnt, urlString, DateTime.Now.Add(timeSpanToStart),
        DateTime.Now.Add(timeSpanInterval), ipaddress);

    // Read the policy into a byte buffer.
    byte[] bufferPolicy = Encoding.ASCII.GetBytes(strPolicy);

    // Convert the policy statement to URL-safe base64 encoding and
    // replace unsafe characters + = / with the safe characters - _ ~

    string urlSafePolicy = ToUrlSafeBase64String(bufferPolicy);

    // Initialize the SHA1CryptoServiceProvider object and hash the policy data.
    byte[] bufferPolicyHash;
    using (SHA1CryptoServiceProvider cryptoSHA1 =
        new SHA1CryptoServiceProvider())
    {
        bufferPolicyHash = cryptoSHA1.ComputeHash(bufferPolicy);

        // Initialize the RSACryptoServiceProvider object.
        RSACryptoServiceProvider providerRSA = new RSACryptoServiceProvider();
        XmlDocument xmlPrivateKey = new XmlDocument();

        // Load PrivateKey.xml, which you created by converting your
        // .pem file to the XML format that the .NET framework uses.
        // Several tools are available.
        xmlPrivateKey.Load("PrivateKey.xml");

        // Format the RSACryptoServiceProvider providerRSA
        // and create the signature.
        providerRSA.FromXmlString(xmlPrivateKey.InnerXml);
        RSAPKCS1SignatureFormatter RSAFormatter =
            new RSAPKCS1SignatureFormatter(providerRSA);
        RSAFormatter.SetHashAlgorithm("SHA1");
        byte[] signedHash = RSAFormatter.CreateSignature(bufferPolicyHash);

        // Convert the signed policy to URL-safe base64 encoding and
        // replace unsafe characters + = / with the safe characters - _ ~
        string strSignedPolicy = ToUrlSafeBase64String(signedHash);

        return urlString +
            "?Policy=" +
            urlSafePolicy +
            "&Signature=" +
```

```
            strSignedPolicy +
            "&Key-Pair-Id=" +
            PrivateKeyId;
    }
}
```

Example 서명 생성을 위한 유틸리티 메서드

다음 메서드는 파일에서 정책 설명을 가져오고 서명 생성을 위해 시간 간격을 구문 분석합니다.

```
public static string CreatePolicyStatement(string policyStmnt,
   string resourceUrl,
   DateTime startTime,
   DateTime endTime,
   string ipAddress)

{
   // Create the policy statement.
   FileStream streamPolicy = new FileStream(policyStmnt, FileMode.Open, FileAccess.Read);
   using (StreamReader reader = new StreamReader(streamPolicy))
   {
      string strPolicy = reader.ReadToEnd();

      TimeSpan startTimeSpanFromNow = (startTime - DateTime.Now);
      TimeSpan endTimeSpanFromNow = (endTime - DateTime.Now);
      TimeSpan intervalStart =
         (DateTime.UtcNow.Add(startTimeSpanFromNow)) -
         new DateTime(1970, 1, 1, 0, 0, 0, DateTimeKind.Utc);
      TimeSpan intervalEnd =
         (DateTime.UtcNow.Add(endTimeSpanFromNow)) -
         new DateTime(1970, 1, 1, 0, 0, 0, DateTimeKind.Utc);

      int startTimestamp = (int)intervalStart.TotalSeconds; // START_TIME
      int endTimestamp = (int)intervalEnd.TotalSeconds;  // END_TIME

      if (startTimestamp > endTimestamp)
         return "Error!";

      // Replace variables in the policy statement.
      strPolicy = strPolicy.Replace("RESOURCE", resourceUrl);
      strPolicy = strPolicy.Replace("START_TIME", startTimestamp.ToString());
      strPolicy = strPolicy.Replace("END_TIME", endTimestamp.ToString());
      strPolicy = strPolicy.Replace("IP_ADDRESS", ipAddress);
      strPolicy = strPolicy.Replace("EXPIRES", endTimestamp.ToString());
      return strPolicy;
   }
}

public static TimeSpan GetDuration(string units, string numUnits)
{
   TimeSpan timeSpanInterval = new TimeSpan();
   switch (units)
   {
      case "seconds":
         timeSpanInterval = new TimeSpan(0, 0, 0, int.Parse(numUnits));
         break;
      case "minutes":
         timeSpanInterval = new TimeSpan(0, 0, int.Parse(numUnits), 0);
         break;
      case "hours":
         timeSpanInterval = new TimeSpan(0, int.Parse(numUnits), 0 ,0);
         break;
      case "days":
         timeSpanInterval = new TimeSpan(int.Parse(numUnits),0 ,0 ,0);
```

```
            break;
         default:
            Console.WriteLine("Invalid time units;" +
               "use seconds, minutes, hours, or days");
            break;
      }
      return timeSpanInterval;
   }

   private static TimeSpan GetDurationByUnits(string durationUnits,
      string startIntervalFromNow)
   {
      switch (durationUnits)
      {
         case "seconds":
            return new TimeSpan(0, 0, int.Parse(startIntervalFromNow));
         case "minutes":
            return new TimeSpan(0, int.Parse(startIntervalFromNow), 0);
         case "hours":
            return new TimeSpan(int.Parse(startIntervalFromNow), 0, 0);
         case "days":
            return new TimeSpan(int.Parse(startIntervalFromNow), 0, 0, 0);
         default:
            return new TimeSpan(0, 0, 0, 0);
      }
   }

   public static string CopyExpirationTimeFromPolicy(string policyStatement)
   {
      int startExpiration = policyStatement.IndexOf("EpochTime");
      string strExpirationRough = policyStatement.Substring(startExpiration +
         "EpochTime".Length);
      char[] digits = { '0', '1', '2', '3', '4', '5', '6', '7', '8', '9' };

      List<char> listDigits = new List<char>(digits);
      StringBuilder buildExpiration = new StringBuilder(20);

      foreach (char c in strExpirationRough)
      {
         if (listDigits.Contains(c))
            buildExpiration.Append(c);
      }
      return buildExpiration.ToString();
   }
```

다음 사항도 참조하십시오.

- Perl을 사용한 URL 서명 생성 (p. 156)
- PHP를 사용한 URL 서명 생성 (p. 164)
- Java를 사용한 URL 서명 생성 (p. 174)
- 비공개 콘텐츠 구성을 위한 도구 및 코드 예제 (p. 454)

Java를 사용한 URL 서명 생성

다음 코드 예제 외에도 Java용 AWS SDK(버전 1)의 `CloudFrontUrlSigner` 유틸리티 클래스를 사용하여 CloudFront 서명 URL (p. 124)을 만들 수 있습니다.

> **Note**
>
> 서명된 URL 생성은 CloudFront를 통해 프라이빗 콘텐츠를 제공 (p. 115)하는 프로세스의 한 부분에 불과합니다. 전체 프로세스에 대한 자세한 내용은 서명된 URL 사용 (p. 124)을 참조하십시오.

다음 예제는 CloudFront 서명 URL을 만드는 방법을 보여줍니다. PEM에서 DER 형식으로 프라이빗 키를 변환해야 하여 이를 사용할 Java를 구현해야 합니다.

Example Java 정책 및 서명 암호화 메서드

```
// Signed URLs for a private distribution
// Note that Java only supports SSL certificates in DER format,
// so you will need to convert your PEM-formatted file to DER format.
// To do this, you can use openssl:
// openssl pkcs8 -topk8 -nocrypt -in origin.pem -inform PEM -out new.der
//    -outform DER
// So the encoder works correctly, you should also add the bouncy castle jar
// to your project and then add the provider.

Security.addProvider(new org.bouncycastle.jce.provider.BouncyCastleProvider());

String distributionDomain = "a1b2c3d4e5f6g7.cloudfront.net";
String privateKeyFilePath = "/path/to/rsa-private-key.der";
String s3ObjectKey = "s3/object/key.txt";
String policyResourcePath = "https://" + distributionDomain + "/" + s3ObjectKey;

// Convert your DER file into a byte array.

byte[] derPrivateKey = ServiceUtils.readInputStreamToBytes(new
    FileInputStream(privateKeyFilePath));

// Generate a "canned" signed URL to allow access to a
// specific distribution and file

String signedUrlCanned = CloudFrontService.signUrlCanned(
    "https://" + distributionDomain + "/" + s3ObjectKey, // Resource URL or Path
    keyPairId,     // Certificate identifier,
                   // an active trusted signer for the distribution
    derPrivateKey, // DER Private key data
    ServiceUtils.parseIso8601Date("2011-11-14T22:20:00.000Z") // DateLessThan
    );
System.out.println(signedUrlCanned);

// Build a policy document to define custom restrictions for a signed URL.

String policy = CloudFrontService.buildPolicyForSignedUrl(
    // Resource path (optional, can include '*' and '?' wildcards)
    policyResourcePath,
    // DateLessThan
    ServiceUtils.parseIso8601Date("2011-11-14T22:20:00.000Z"),
    // CIDR IP address restriction (optional, 0.0.0.0/0 means everyone)
    "0.0.0.0/0",
    // DateGreaterThan (optional)
    ServiceUtils.parseIso8601Date("2011-10-16T06:31:56.000Z")
    );

// Generate a signed URL using a custom policy document.

String signedUrl = CloudFrontService.signUrl(
    // Resource URL or Path
    "https://" + distributionDomain + "/" + s3ObjectKey,
    // Certificate identifier, an active trusted signer for the distribution
    keyPairId,
    // DER Private key data
    derPrivateKey,
    // Access control policy
    policy
    );
System.out.println(signedUrl);
```

추가 참고:

- Perl을 사용한 URL 서명 생성 (p. 156)
- PHP를 사용한 URL 서명 생성 (p. 164)
- C# 및 .NET Framework를 사용한 URL 서명 생성 (p. 169)
- 비공개 콘텐츠 구성을 위한 도구 및 코드 예제 (p. 454)

원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한

Amazon S3 버킷에서 제공하는 콘텐츠에 대한 액세스를 제한하려면 CloudFront 서명된 URL 또는 서명된 쿠키를 만듭니다. 그런 다음 원본 액세스 ID(OAI)라는 특수 CloudFront 사용자를 만들어 배포와 연결합니다. 그런 다음 CloudFront가 OAI를 사용하여 사용자에 액세스하고 파일을 제공할 수 있지만, 사용자는 S3 버킷에 대한 직접 URL을 사용하여 파일에 액세스할 수 없도록 권한을 구성합니다. 이 단계를 수행하면 CloudFront를 통해 제공하는 파일에 대한 액세스를 안전하게 유지할 수 있습니다.

일반적으로 Amazon S3 버킷을 CloudFront 배포의 오리진으로 사용하는 경우 모든 사용자에게 해당 버킷의 파일에 액세스하도록 허용하거나 액세스를 제한할 수 있습니다 예를 들어 CloudFront 서명된 URL 또는 서명된 쿠키를 사용해 액세스를 제한하는 경우 사용자가 단순히 파일에 대한 직접 URL을 사용하여 파일을 보는 것도 원치 않을 것입니다. 대신, 보호가 작동하도록 CloudFront URL을 사용해야만 파일에 액세스하도록 허용할 수 있습니다. 서명된 URL 및 서명된 쿠키 사용에 대한 자세한 내용은 서명된 URL과 서명된 쿠키를 사용하여 프라이빗 콘텐츠 제공 (p. 115) 단원을 참조하십시오.

이 주제는 OAI를 설정하고 S3 파일에 대한 액세스를 안전하게 유지하기 위한 권한을 부여하는 방법을 자세히 설명합니다.

> **Important**
>
> 웹 사이트 엔드포인트로 구성된 Amazon S3 버킷을 사용하는 경우 이 버킷을 CloudFront에서 사용자 지정 오리진으로 설정해야 하며, 이 주제에서 설명하는 원본 액세스 ID 기능은 사용할 수 없습니다. 그러나, 사용자 지정 헤더를 설정하고 해당 헤더가 필요하도록 오리진을 구성하여 사용자 지정 오리진의 콘텐츠에 대한 액세스를 제한할 수 있습니다. 자세한 내용은 사용자 지정 오리진의 파일에 대한 액세스 제한 (p. 117) 단원을 참조하십시오.

주제

- OAI 설정 개요 (p. 176)
- CloudFront OAI 생성 및 배포에 추가 (p. 177)
- OAI에 Amazon S3 버킷 내 파일 읽기 권한 부여 (p. 179)
- 서명 버전 4 인증만 지원하는 Amazon S3 리전에서 OAI 사용 (p. 182)

OAI 설정 개요

Amazon S3 버킷을 CloudFront 배포의 오리진으로 처음 설정하면 모든 사용자에게 버킷의 파일에 대한 권한을 부여하게 됩니다. 이렇게 하면 누구나 CloudFront를 통해 또는 Amazon S3 URL을 사용해 파일에 액세스할 수 있습니다. CloudFront는 Amazon S3 URL을 공개하지 않지만, 애플리케이션이 Amazon S3에서 직접 파일을 제공하거나 누군가가 Amazon S3의 특정 파일에 대한 직접 링크를 제공하는 경우 사용자가 그러한 URL을 가지게 될 수 있습니다.

CloudFront 서명된 URL 또는 서명된 쿠키를 사용하여 Amazon S3 버킷의 파일에 대한 액세스를 제한하는 경우에는 Amazon S3 URL로 Amazon S3 객체에 액세스하는 사용자도 차단하는 것이 좋습니다. Amazon S3에서 직접 파일에 액세스하는 사용자는 CloudFront 서명된 URL 또는 서명된 쿠키를 이용한 제어를 우회할

수 있습니다. 예를 들어, 사용자의 콘텐츠 액세스를 차단할 날짜와 시간에 대한 제어 및 콘텐츠에 액세스할 수 있는 IP 주소 제어 등을 우회합니다. 또한 사용자가 CloudFront을(를) 통해서도 파일에 액세스하고 직접 Amazon S3 URL을 통해서도 액세스하는 경우 CloudFront 액세스 로그가 불완전하여 유용성이 떨어집니다.

URL 서명 여부와 관계없이 오직 CloudFront URL로만 파일에 액세스할 수 있게 하려면 다음 작업을 수행하십시오.

1. 특별한 CloudFront 사용자인 원본 액세스 ID를 만들고 배포와 연결합니다. 웹 배포의 경우, 원본 액세스 ID를 오리진과 연결하여 Amazon S3 콘텐츠 중 일부 또는 전부를 보호할 수 있습니다. 또한 원본 액세스 ID를 만들고 배포를 생성한 후 배포에 추가할 수 있습니다. 자세한 내용은 CloudFront OAI 생성 및 배포에 추가 (p. 177) 단원을 참조하십시오.
2. 원본 액세스 ID만 읽기 권한(또는 읽기 및 다운로드 권한)을 가지도록 Amazon S3 버킷 또는 버킷의 파일에 대한 권한을 변경합니다. 사용자가 CloudFront을(를) 통해 Amazon S3 파일에 액세스할 때, CloudFront 원본 액세스 ID가 사용자를 대신해 파일을 가져옵니다. 사용자가 Amazon S3 URL을 사용하여 직접 파일을 요청하면 액세스가 거부됩니다. Amazon S3 버킷에 있는 파일에 액세스할 수 있는 권한이 원본 액세스 ID에는 있지만 사용자에게는 없습니다. 자세한 내용은 OAI에 Amazon S3 버킷 내 파일 읽기 권한 부여 (p. 179) 단원을 참조하십시오.

   Tip

   이제 막 S3 버킷의 프라이빗 콘텐츠로 CloudFront를 설정하기 시작한 경우 프로세스의 종합적 보기에 대한 How to Set Up and Serve Private Content Using S3 and Amazon CloudFront 블로그 게시물을 읽는 것이 유용할 수 있습니다.

CloudFront OAI 생성 및 배포에 추가

AWS 계정 하나는 최대 100개의 CloudFront 원본 액세스 ID(OAI) (p. 447)를 가질 수 있습니다. 그러나 OAI 하나를 원하는 수만큼의 배포에 추가할 수 있기 때문에 보통 하나면 충분합니다.

OAI를 만들지 않은 상태에서 배포를 만들 때 배포에 OAI를 추가해야 한다면 CloudFront 콘솔 또는 CloudFront API를 사용하여 OAI를 만들고 추가할 수 있습니다.

- CloudFront 콘솔을 사용하는 경우 – OAI를 만드는 동시에 이를 배포에 추가할 수 있습니다. 단계별 지침은 OAI 생성 및 배포에 추가 (p. 177)를 참조하십시오.
- CloudFront API를 사용하는 경우 – OAI를 만든 후 배포에 추가합니다. 단계별 지침은 다음을 참조하십시오.
  - CloudFront API를 사용하여 OAI 생성 (p. 178)
  - CloudFront API를 사용하여 배포에 OAI 추가 (p. 178)

  Note

  OAI를 만들려면 CloudFront 콘솔 또는 CloudFront API 버전 2009-09-09 이상을 사용해야 합니다.

OAI 생성 및 배포에 추가

배포를 생성할 때 OAI를 만들지 않은 경우 다음 작업을 수행합니다.

CloudFront 콘솔을 사용하여 CloudFront OAI를 만들려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. S3 오리진이 있는 배포의 ID를 선택합니다.
3. Origins and Origin Groups(오리진 및 오리진 그룹) 탭을 선택합니다.

4. 오리진 옆의 확인란을 선택한 다음 Edit(편집)을 선택합니다.
5. Restrict Bucket Access(버킷 액세스 제한)에서 예를 선택합니다.
6. 사용할 OAI가 이미 있는 경우 Use an Existing Identity(기존 자격 증명 사용)를 선택합니다. 그런 다음 Your Identities(사용자 자격 증명) 목록에서 OAI를 선택합니다.

   Note

   OAI가 이미 있는 경우 이것을 재사용하여 유지 관리를 간소화하는 것이 좋습니다.

   OAI를 만들려면 Create a New Identity(새 자격 증명 만들기)를 선택합니다. Comment(설명) 필드의 버킷 이름을 사용자 지정 설명으로 바꿀 수 있습니다.
7. CloudFront가 Origin Domain Name(오리진 도메인 이름)에 지정된 Amazon S3 버킷의 파일에 대한 읽기 권한을 자동으로 OAI에 부여하도록 하려면 Yes, Update Bucket Policy(예, 버킷 정책 업데이트)를 선택합니다.

   Important

   Yes, Update Bucket Policy(예, 버킷 정책 업데이트)를 선택하면 CloudFront는 버킷 권한을 업데이트하여 지정된 OAI에 버킷 파일 읽기 권한을 부여합니다. 그러나 CloudFront가 기존 권한을 제거하지는 않습니다. 현재 Amazon S3 URL을 통해 버킷 파일에 액세스할 수 있는 사용자는 CloudFront에서 버킷 권한을 업데이트한 후에도 동일한 권한을 가집니다. 기존 버킷 권한을 확인 또는 제거하려면 Amazon S3에서 제공하는 메서드를 사용하십시오.

   Amazon S3 버킷 권한을 수동으로 업데이트하려면 No, I Will Update Permissions(아니요, 권한을 업데이트하겠습니다)를 선택합니다. 자세한 내용은 OAI에 Amazon S3 버킷 내 파일 읽기 권한 부여 (p. 179) 단원을 참조하십시오.
8. 예, 편집합니다를 선택합니다.
9. 오리진이 여러 개 있을 경우 이 절차를 반복하여 각 오리진에 대한 OAI를 추가합니다.

CloudFront API를 사용하여 OAI 생성

원본 액세스 ID가 이미 있어서 새로 만들지 않고 기존 ID를 재사용하려는 경우, CloudFront API를 사용하여 배포에 OAI 추가 (p. 178) 단원으로 건너뛰세요.

CloudFront API를 사용하여 CloudFront OAI를 만들려면 `POST Origin Access Identity` API 작업을 사용합니다. 응답에는 새 OAI에 대한 `Id` 및 `S3CanonicalUserId`가 포함됩니다. 프로세스 후반에 이러한 값을 사용할 것이므로 기록해 두십시오.

- Id 요소 – `Id` 요소의 값을 사용하여 OAI를 배포와 연결합니다.
- S3CanonicalUserId 요소 – Amazon S3 객체 ACL을 사용하여 OAI에 Amazon S3 객체에 대한 액세스 권한을 부여할 때 `S3CanonicalUserId` 요소의 값을 사용합니다.

자세한 내용은 Amazon CloudFront API Reference의 CreateCloudFrontOriginAccessIdentity를 참조하십시오.

CloudFront API를 사용하여 배포에 OAI 추가

CloudFront API를 사용하여 CloudFront OAI를 기존 배포에 추가하거나 OAI를 포함하는 새로운 배포를 만들 수 있습니다. 두 경우 모두 `OriginAccessIdentity` 요소를 포함합니다. 이 요소는 OAI를 만든 후 `Id` API 작업에서 반환된 `POST Origin Access Identity` 요소 값을 포함합니다. `OriginAccessIdentity` 요소를 한 개 이상의 오리진에 추가할 수 있습니다.

Amazon CloudFront API Reference에서 다음 주제를 참조하십시오.

- 새로운 웹 배포 생성 – CreateDistribution

- 기존 웹 배포 업데이트 – UpdateDistribution

OAI에 Amazon S3 버킷 내 파일 읽기 권한 부여

배포를 만들거나 업데이트할 때 OAI(원본 액세스 ID)를 추가하고 Amazon S3 버킷 정책을 자동으로 업데이트하여 버킷에 액세스할 수 있는 권한을 OAI에 부여합니다. 또는 버킷 정책을 수동으로 생성 또는 업데이트하거나 버킷의 개별 파일에 대한 액세스를 제어하는 객체 ACL을 사용할 수 있습니다.

어떤 방법을 사용하든 권한을 검토하여 다음 사항을 확인하십시오.

- CloudFront를 통해 파일을 요청하는 최종 사용자를 대신하여 CloudFront OAI가 버킷의 파일에 액세스할 수 있습니다.
- 최종 사용자는 Amazon S3 URL을 사용하여 CloudFront 외부의 파일에 액세스할 수 없습니다.

> **Important**
>
> CloudFront가 지원하는 모든 HTTP 메서드를 수락하고 전달하도록 CloudFront를 구성하는 경우 CloudFront OAI에 원하는 권한을 부여해야 합니다. 예를 들어 `DELETE` 메서드를 사용하는 요청을 수락하고 전달하도록 CloudFront를 구성하는 경우, 삭제되면 안 되는 파일을 최종 사용자가 삭제하지 못하도록 `DELETE` 요청을 적절히 처리할 수 있는 버킷 정책 또는 객체 ACL을 구성합니다.

다음을 참조하십시오.

- 객체 ACL보다 Amazon S3 버킷 정책을 업데이트하는 것이 더 쉽습니다. 권한을 업데이트하지 않고도 파일을 버킷에 추가할 수 있기 때문입니다. 그러나 각 개별 파일에 권한을 부여하기 때문에 ACL은 더 세부적인 제어가 가능합니다.
- 기본적으로 Amazon S3 버킷과 그 안에 있는 모든 파일은 비공개입니다. 버킷을 만든 AWS 계정만 안에 있는 파일을 읽거나 쓸 수 있는 권한이 있습니다.
- 다른 AWS 계정이 버킷에 파일을 업로드하는 경우, 그 계정이 해당 파일의 소유자가 됩니다. 버킷 정책은 버킷 소유자가 소유한 파일에만 적용됩니다. 즉, 다른 계정에서 내 버킷에 파일을 업로드하는 경우 내가 OAI에 대해 생성한 버킷 정책은 해당 파일에 대해 평가되지 않습니다. 이 경우 객체 ACL을 사용하여 OAI에 권한을 부여합니다.
- OAI를 기존 배포에 추가하는 경우, 파일을 CloudFront 외부에 공개적으로 이용할 수 없도록 버킷 정책 또는 객체 ACL을 수정해야 합니다.
- 하나 이상의 안전한 관리자 계정에 권한을 추가로 부여하여 Amazon S3 버킷의 콘텐츠를 계속 업데이트할 수 있습니다.

> **Important**
>
> Amazon S3 권한에 대한 변경 사항을 저장한 후 변경 사항이 적용되기까지 약간 지연될 수 있습니다. 변경 사항이 적용되기 전에 버킷의 파일에 액세스하려 하면 “권한 거부” 오류가 발생할 수 있습니다.

Amazon S3 버킷 정책 사용

다음과 같은 방법으로 버킷 정책을 만들거나 업데이트하여 Amazon S3 버킷의 파일에 대한 액세스 권한을 CloudFront OAI에 부여할 수 있습니다.

- Amazon S3 콘솔에서 Amazon S3 버킷의 권한 탭 사용
- Amazon S3 API에서 PutBucketPolicy 사용
- CloudFront 콘솔 사용 OAI를 CloudFront 콘솔의 오리진 설정에 추가할 때 Yes, Update Bucket Policy(예, 버킷 정책을 업데이트합니다.)를 선택하여 사용자를 대신해 버킷 정책을 업데이트하도록 CloudFront에 지시할 수 있습니다.

버킷 정책을 수동으로 업데이트하는 경우 다음 사항을 확인하십시오.

- 정책에서 올바른 OAI를 `Principal`로 지정합니다.
- 최종 사용자를 대신해 객체에 액세스하는 데 필요한 권한을 OAI에 부여합니다.

자세한 내용은 다음 단원을 참조하십시오.

정책에서 OAI를 `Principal`로 지정

Amazon S3 버킷 정책에서 OAI를 `Principal`로 지정하려면 OAI의 ID를 사용합니다. 다음 예를 참조하십시오.

```
"Principal": {
    "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access Identity EH1HDMB1FH2TC"
}
```

앞의 예제를 사용하려면 *`EH1HDMB1FH2TC`*를 OAI의 ID로 대체합니다. OAI의 ID를 찾으려면 CloudFront 콘솔에서 원본 액세스 ID 페이지를 참조하거나 CloudFront API에서 ListCloudFrontOriginAccessIdentities를 사용합니다.

Amazon S3 정식 ID를 사용하여 OAI를 `Principal`로 지정할 수도 있습니다. 다음 예를 참조하십시오.

```
"Principal": {
    "CanonicalUser": "79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be"
}
```

앞의 예제를 사용하려면 *`79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be`*를 OAI의 정식 ID로 대체합니다. OAI의 ID를 찾는 것과 동일한 방법으로 OAI의 정식 ID를 찾을 수 있습니다.

> **Note**
>
> OAI의 ID를 사용하여 `Principal`로 지정할 때의 한 가지 이점은 버킷 정책이 누구에게 액세스 권한을 부여하는지 보다 쉽게 이해할 수 있다는 것입니다. Amazon S3 정식 ID는 다른 종류의 AWS 자격 증명을 참조할 수 있지만 CloudFront OAI는 참조할 수 없으며 정식 ID가 참조하는 자격 증명을 확인하기가 어려울 수 있습니다. OAI의 ID를 사용하면 버킷 정책을 보다 쉽게 이해할 수 있습니다. 또한 버킷 정책에서 OAI의 정식 ID를 사용하는 경우 AWS는 이 정식 ID를 OAI의 ID로 대체합니다. 즉, OAI의 정식 ID를 지정하는 정책을 작성한 후 나중에 동일한 정책을 보면, 이 정식 ID가 해당 OAI ID로 대체되었음을 확인할 수 있습니다. 이러한 이유로 OAI의 ID를 사용하여 정책을 작성하는 것이 합리적일 수 있습니다.

정책에서 OAI 권한 부여

Amazon S3 버킷의 객체에 액세스하는 데 필요한 권한을 OAI에 부여하려면 특정 Amazon S3 API 작업과 관련된 권한 키워드를 사용합니다. 예를 들어 `s3:GetObject` 권한을 통해 OAI가 버킷의 객체를 읽을 수 있습니다. 자세한 내용은 다음 단원의 예제를 참조하거나 Amazon Simple Storage Service 개발자 가이드의 정책에서 권한 지정을 참조하십시오.

Amazon S3 버킷 정책 예제

다음 예제에서는 CloudFront OAI에 액세스 권한을 부여하는 Amazon S3 버킷 정책을 보여줍니다. 이러한 예제를 사용하려면

- *`EH1HDMB1FH2TC`*를 OAI의 ID로 대체합니다. OAI의 ID를 찾으려면 CloudFront 콘솔에서 원본 액세스 ID 페이지를 참조하거나 CloudFront API에서 ListCloudFrontOriginAccessIdentities를 사용합니다.
- *`aws-example-bucket`*을 Amazon S3 버킷의 이름으로 대체합니다.

Example OAI에 읽기 액세스 권한을 부여하는 Amazon S3 버킷 정책

다음 예제에서는 OAI가 지정된 버킷(`s3:GetObject`)의 객체를 읽을 수 있도록 허용합니다.

```
{
    "Version": "2012-10-17",
    "Id": "PolicyForCloudFrontPrivateContent",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access
 Identity EH1HDMB1FH2TC"
            },
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::aws-example-bucket/*"
        }
    ]
}
```

Example OAI에 읽기 및 쓰기 액세스 권한을 부여하는 Amazon S3 버킷 정책

다음 예제에서는 OAI가 지정된 버킷(`s3:GetObject` 및 `s3:PutObject`)의 객체를 읽고 쓸 수 있도록 허용합니다. 이렇게 하면 최종 사용자가 CloudFront를 통해 Amazon S3 버킷에 파일을 업로드할 수 있습니다.

```
{
    "Version": "2012-10-17",
    "Id": "PolicyForCloudFrontPrivateContent",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access
 Identity EH1HDMB1FH2TC"
            },
            "Action": [
                "s3:GetObject",
                "s3:PutObject"
            ],
            "Resource": "arn:aws:s3:::aws-example-bucket/*"
        }
    ]
}
```

Amazon S3 객체 ACL 업데이트

다음과 같은 방법으로 파일의 ACL을 만들거나 업데이트하여 Amazon S3 버킷의 파일에 대한 액세스 권한을 CloudFront OAI에 부여할 수 있습니다.

- Amazon S3 콘솔에서 Amazon S3 객체의 권한 탭 사용
- Amazon S3 API에서 PutObjectAcl 사용

ACL을 사용하여 OAI에 액세스 권한을 부여하는 경우 Amazon S3 정식 사용자 ID를 사용하여 OAI를 지정해야 합니다. 이는 CloudFront 콘솔의 원본 액세스 ID 페이지에 있는 Amazon S3 정식 사용자 ID 값입니다. CloudFront API를 사용하는 경우, OAI를 만들 때 반환된 `S3CanonicalUserId` 요소의 값을 사용하거나 CloudFront API에서 ListCloudFrontOriginAccessIdentities를 호출합니다.

Amazon S3 객체 ACL에 대한 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 ACL을 사용한 액세스 관리 단원을 참조하십시오.

서명 버전 4 인증만 지원하는 Amazon S3 리전에서 OAI 사용

새 Amazon S3 리전에서는 인증된 요청에 대해 서명 버전 4를 사용해야 합니다. (각 Amazon S3 리전에서 지원되는 서명 버전은 Amazon Web Services 일반 참조의 리전 및 엔드포인트 단원에서 Amazon Simple Storage Service(S3)를 참조하십시오.) 그러나 원본 액세스 ID를 만들어 이를 CloudFront 배포에 추가하는 경우, CloudFront는 일반적으로 Amazon S3 버킷 파일 요청에 대한 인증에 서명 버전 4를 사용합니다. 원본 액세스 ID를 사용 중인데 해당 버킷이 인증에 대해 서명 버전 4를 요구하는 리전에 속하는 경우 다음에 유의해야 합니다.

- `DELETE`, `GET`, `HEAD`, `OPTIONS` 및 `PATCH` 요청은 자격과 관계없이 지원됩니다.
- `PUT` 요청을 CloudFront에 제출하여 Amazon S3 버킷에 파일을 업로드하려면 `x-amz-content-sha256` 헤더를 요청에 추가해야 하고, 헤더 값에 요청 본문의 SHA256 해시가 포함되어야 합니다. 자세한 내용은 Amazon Simple Storage Service API Reference의 일반 요청 헤더 페이지에서 `x-amz-content-sha256` 헤더에 대한 설명서를 참조하십시오.
- `POST` 요청은 지원되지 않습니다.

AWS AWS WAF를 사용하여 콘텐츠에 대한 액세스 제어

AWS WAF는 CloudFront에 전달되는 HTTP 및 HTTPS 요청을 모니터링할 수 있게 해주고 콘텐츠에 대한 액세스를 제어할 수 있게 해주는 웹 애플리케이션 방화벽입니다. 요청이 허용되는 IP 주소나 쿼리 문자열의 값으로부터 지정하는 조건 등 지정하는 조건에 따라, CloudFront는 요청된 콘텐츠나 HTTP 403 상태 코드(금지됨)로 요청에 응답합니다. 또한 요청이 차단될 때 사용자 지정 오류 페이지를 반환하도록 CloudFront를 구성할 수 있습니다. AWS WAF에 대한 자세한 내용은 AWS WAF 개발자 안내서를 참조하십시오.

AWS WAF 웹 액세스 통제 목록(웹 ACL)을 생성한 후, 웹 배포를 생성 또는 업데이트하고 배포를 웹 ACL과 연결합니다. 같은 웹 ACL 또는 다른 웹 ACL로 CloudFront 배포를 원하는 만큼 연결할 수 있습니다. 웹 배포 만들기 및 웹 ACL과 연결에 대한 자세한 내용은 배포 만들기 (p. 35)을 참조하십시오.

웹 ACL 및 기존 배포를 연결 또는 연결 해제하거나 배포와 연결된 웹 ACL을 변경하려면 다음 절차를 수행합니다.

CloudFront 콘솔을 사용하여 AWS WAF 웹 ACL 및 기존 CloudFront 배포를 연결 또는 연결 해제하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 업데이트하려는 배포의 ID를 선택합니다.
3. [General] 탭에서 [Edit]를 선택합니다.
4. Distribution Settings(배포 설정) 페이지의 AWS WAF 웹 ACL 목록에서 이 배포와 연결하려는 웹 ACL을 선택합니다.

   모든 웹 ACL에서 배포를 연결 해제하려면 없음을 선택합니다. 다른 웹 ACL에서 배포를 연결하려면 새 웹 ACL을 선택합니다.
5. 예, 편집합니다를 선택합니다.
6. AWS WAF 웹 ACL과의 연결을 추가하거나, 삭제하거나, 변경하려는 다른 배포가 있다면 2~5단계를 반복합니다.
7. 설정을 변경한 후 변경 사항을 업데이트한 배포에 대한 상태 열의 값이 진행 중으로 변경되고 CloudFront는 엣지 로케이션으로 변경 사항을 전파합니다. 배포의 상태가 배포 완료로 변경되면 배포는 요청을 처리할 때 AWS WAF를 사용할 수 있습니다. (배포에 대한 상태 열의 값도 활성이 되어야 합니다.) 배포에 마지막 변경 사항을 저장한 후 15분도 걸리지 않습니다.

Note

AWS Firewall Manager는 계정과 애플리케이션 전체에 대한 AWS WAF 규칙을 좀 더 쉽게 중앙에서 구성 및 관리할 수 있는 보안 관리 서비스입니다. Firewall Manager를 사용하면 AWS Organizations 내의 계정에서 CloudFront 배포용 AWS WAF 규칙을 롤아웃할 수 있습니다. 자세한 내용은 AWS Firewall Manager 개발자 안내서를 참조하십시오.

콘텐츠의 지리적 배포 제한

지리적 차단이라고도 하는 지리적 제한을 사용하여 특정 지리적 위치에 있는 사용자가 CloudFront 웹 배포를 통해 배포한 콘텐츠에 액세스하는 것을 차단할 수 있습니다. 지리적 제한을 사용하는 데에는 다음 두 가지 옵션이 있습니다.

- CloudFront 지리적 제한 기능을 사용합니다. 이 옵션을 사용하면 배포와 연결된 파일 전체에 대한 액세스를 제한하고 국가 수준에서 액세스를 제한합니다.
- 타사 지리적 위치 서비스를 사용합니다. 이 옵션을 사용하면 배포와 연결된 파일의 하위 집합에 대한 액세스를 제한하거나 국가 수준이 아닌 더욱 세분화된 범위에서 액세스를 제한합니다.

주제

- CloudFront 지리적 제한 사용 (p. 183)
- 타사 지리적 위치 서비스 사용 (p. 184)

CloudFront 지리적 제한 사용

사용자가 콘텐츠를 요청할 경우 CloudFront에서는 대개 이 사용자가 있는 위치와 관계없이 요청된 콘텐츠를 제공합니다. 특정 국가의 사용자들이 콘텐츠에 액세스하는 것을 방지해야 할 경우, 다음 중 하나를 수행하도록 CloudFront 지리적 제한 기능을 사용할 수 있습니다.

- 사용자가 승인된 국가의 화이트리스트에 있는 국가 중 하나에 있는 경우에만 액세스를 허용합니다.
- 사용자가 금지된 국가의 블랙리스트에 있는 국가 중 하나에 있는 경우 액세스를 금지합니다.

예를 들어, 특정 국가로부터 요청이 온 경우 저작권상 이유로 콘텐츠를 배포할 권한은 없으며 CloudFront 지리적 제한을 사용하여 이 요청을 차단할 수는 있습니다.

Note

CloudFront에서는 타사 GeoIP 데이터베이스를 사용하여 사용자의 위치를 확인합니다. IP 주소 및 국가 간 매핑 정확도는 리전에 따라 다릅니다. 최근 테스트에 따르면 전반적인 정확도는 99.8%입니다. CloudFront에서 사용자 위치를 확인할 수 없는 경우 CloudFront는 사용자가 요청한 콘텐츠를 제공합니다.

지리적 제한이 작동하는 방식은 다음과 같습니다.

1. 리히텐슈타인에서만 콘텐츠를 배포할 권리를 보유하고 있다고 가정합니다. CloudFront 웹 배포를 업데이트하고 리히텐슈타인만 포함된 화이트리스트를 추가합니다. (또는 리히텐슈타인을 제외한 모든 국가가 포함된 블랙리스트를 추가할 수도 있습니다.)
2. 모나코에 있는 사용자가 콘텐츠를 요청하고 DNS는 요청을 이탈리아 밀라노에 있는 CloudFront 엣지 로케이션으로 라우팅합니다.
3. 밀라노의 엣지 로케이션에서는 이 배포를 찾아 모나코에 있는 사용자의 콘텐츠 다운로드가 허용되지 않았음을 확인합니다.

4. CloudFront에서는 HTTP 상태 코드 403(금지됨)을 사용자에게 반환합니다.

선택적으로 CloudFront에서 사용자 지정 오류 메시지를 사용자에게 반환하도록 구성할 수 있으며, CloudFront에서 요청된 파일에 대한 오류 응답을 캐싱하려는 시간(기본값은 5분)을 지정할 수 있습니다. 자세한 내용은 특정 HTTP 상태 코드에 대한 사용자 지정 오류 페이지 만들기 (p. 255) 단원을 참조하십시오.

지리적 제한은 전체 웹 배포에 적용됩니다. 하나의 제한을 콘텐츠의 특정 부분에 적용하고 또 다른 제한을 다른 부분에 적용해야 할 경우(또는 이 부분에 대한 제한이 없는 경우), 별도의 CloudFront 웹 배포를 만들거나 타사 지리적 위치 서비스를 사용해야 합니다.

CloudFront 액세스 로깅을 활성화한 경우, `sc-status`(HTTP 상태 코드)의 값이 `403`인 로그 항목을 검색하여 CloudFront에서 거부된 요청을 식별할 수 있습니다. 그러나, 액세스 로그만 사용해서는 사용자의 위치에 따라 CloudFront에서 거부된 요청과 사용자가 다른 이유로 파일에 대한 액세스 권한이 없어 CloudFront에서 거부된 요청을 구별할 수 없습니다. Digital Element 또는 MaxMind와 같은 타사 지리적 위치 서비스가 있는 경우, 액세스 로그에 있는 `c-ip`(클라이언트 IP) 열의 IP 주소에 따라 요청의 위치를 식별할 수 있습니다. CloudFront 액세스 로그에 대한 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

다음 절차는 CloudFront 콘솔을 사용하여 기존 웹 배포에 지리적 제한을 추가하는 방법을 설명합니다. 콘솔을 사용하여 웹 배포를 생성하는 방법에 대한 자세한 내용은 배포 만들기 (p. 35) 단원을 참조하십시오.

CloudFront 콘솔을 사용하여 CloudFront 웹 배포에 지리적 제한을 추가하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 업데이트하려는 배포를 선택합니다.
3. Distribution Settings(배포 설정) 창에서 Restrictions(제한) 탭을 선택합니다.
4. [Edit]를 선택합니다.
5. 관련 값들을 입력합니다. 자세한 내용은 제한 사항 (p. 57) 단원을 참조하십시오.
6. 예, 편집합니다를 선택합니다.

타사 지리적 위치 서비스 사용

CloudFront 지리적 제한 기능을 통해 지정된 웹 배포를 통해 배포하는 전체 파일에 대해 국가 수준에서 콘텐츠의 배포를 제어할 수 있습니다. 콘텐츠가 배포될 수 있는 지역에 대해 지리적 제한이 있고 이 제한이 국가의 경계를 따르지 않는 경우, 또는 CloudFront를 통해 배포하는 파일 중 일부에 대해서만 액세스를 제한하려는 경우, CloudFront와 타사 지리적 위치 서비스를 조합할 수 있습니다. 이를 통해 국가에 따라서나 도시, 우편번호에 따라서는 물론이고 경도/위도에 따라서 콘텐츠에 대한 액세스를 제어할 수 있습니다.

타사 지리적 위치 서비스를 사용하는 경우 CloudFront 서명된 URL을 사용하는 것이 좋습니다. 이렇게 하면 URL이 더 이상 유효하지 않게 된 이후의 만료 날짜 및 시간을 지정할 수 있습니다. 또한 오리진으로는 Amazon S3 버킷을 사용하는 것이 좋습니다. 이렇게 하면 CloudFront 원본 액세스 ID를 사용하여 사용자가 오리진에서 콘텐츠에 직접 액세스하는 것을 차단할 수 있기 때문입니다. 서명된 URL 및 원본 액세스 ID에 대한 자세한 내용은 서명된 URL과 서명된 쿠키를 사용하여 프라이빗 콘텐츠 제공 (p. 115) 단원을 참조하십시오.

다음 작업 목록에서는 타사 지리적 위치 서비스를 사용하여 파일에 대한 액세스를 제어하는 방법을 설명합니다.

지리적 위치에 따라 CloudFront 배포의 파일에 대한 액세스를 제한하기 위한 작업 목록

1. 지리적 위치 서비스를 통해 계정을 얻습니다.
2. Amazon Simple Storage Service(S3) 버킷에 콘텐츠를 업로드합니다. 자세한 내용은 Amazon S3 설명서를 참조하십시오.
3. 비공개 콘텐츠를 제공하도록 Amazon CloudFront 및 Amazon S3를 구성합니다. 자세한 내용은 서명된 URL과 서명된 쿠키를 사용하여 프라이빗 콘텐츠 제공 (p. 115) 단원을 참조하십시오.

4. 다음을 수행하도록 웹 애플리케이션을 작성합니다.

   a. 각 사용자 요청에 대한 IP 주소를 지리적 위치 서비스에 전송합니다.

   b. 지리적 위치 서비스로부터의 반환 값을 평가하여 CloudFront에서 콘텐츠를 배포하려는 위치에 사용자가 있는지 여부를 확인합니다.

   c. 사용자의 위치에 콘텐츠를 배포할지 여부에 따라 CloudFront 콘텐츠에 대한 서명된 URL을 생성하거나 사용자에게 HTTP 상태 코드 403(금지됨)을 반환합니다. 또는 사용자 지정 오류 메시지를 반환하도록 CloudFront를 구성할 수 있습니다. 자세한 내용은 특정 HTTP 상태 코드에 대한 사용자 지정 오류 페이지 만들기 (p. 255) 단원을 참조하십시오.

   자세한 내용은 사용 중인 지리적 위치 서비스의 설명서를 참조하십시오.

웹 서버 변수를 사용하여 웹 사이트를 방문한 사용자의 IP 주소를 가져올 수 있습니다. 다음 경고를 기록해 두세요.

- 웹 서버가 로드 밸런서를 통해 인터넷에 연결되지 않은 경우, 웹 서버 변수를 사용하여 원격 IP 주소를 가져올 수 있습니다. 그러나, 이 IP 주소는 사용자의 IP 주소와 항상 일치하지는 않으며, 사용자가 인터넷에 연결된 방법에 따라 프록시 서버의 IP 주소가 될 수도 있습니다.
- 웹 서버가 로드 밸런서를 통해 인터넷에 연결된 경우, 웹 서버 변수에는 사용자의 IP 주소가 아닌 로드 밸런서의 IP 주소가 포함될 수 있습니다. 이러한 구성에서는 `X-Forwarded-For` http 헤더의 마지막 IP 주소를 사용하는 것이 좋습니다. 이 헤더는 대개 두 개 이상의 IP 주소를 포함하고 있으며, 이들 중 대부분은 프록시나 로드 밸런서용 IP 주소입니다. 목록의 마지막 IP 주소는 사용자의 지리적 위치와 가장 관련성이 높은 주소입니다.

웹 서버가 로드 밸런서에 연결되지 않은 경우, IP 주소 스푸핑을 방지하기 위해 `X-Forwarded-For` 헤더 대신 웹 서버 변수를 사용하는 것이 좋습니다.

필드 레벨 암호화 사용으로 민감한 데이터 보호

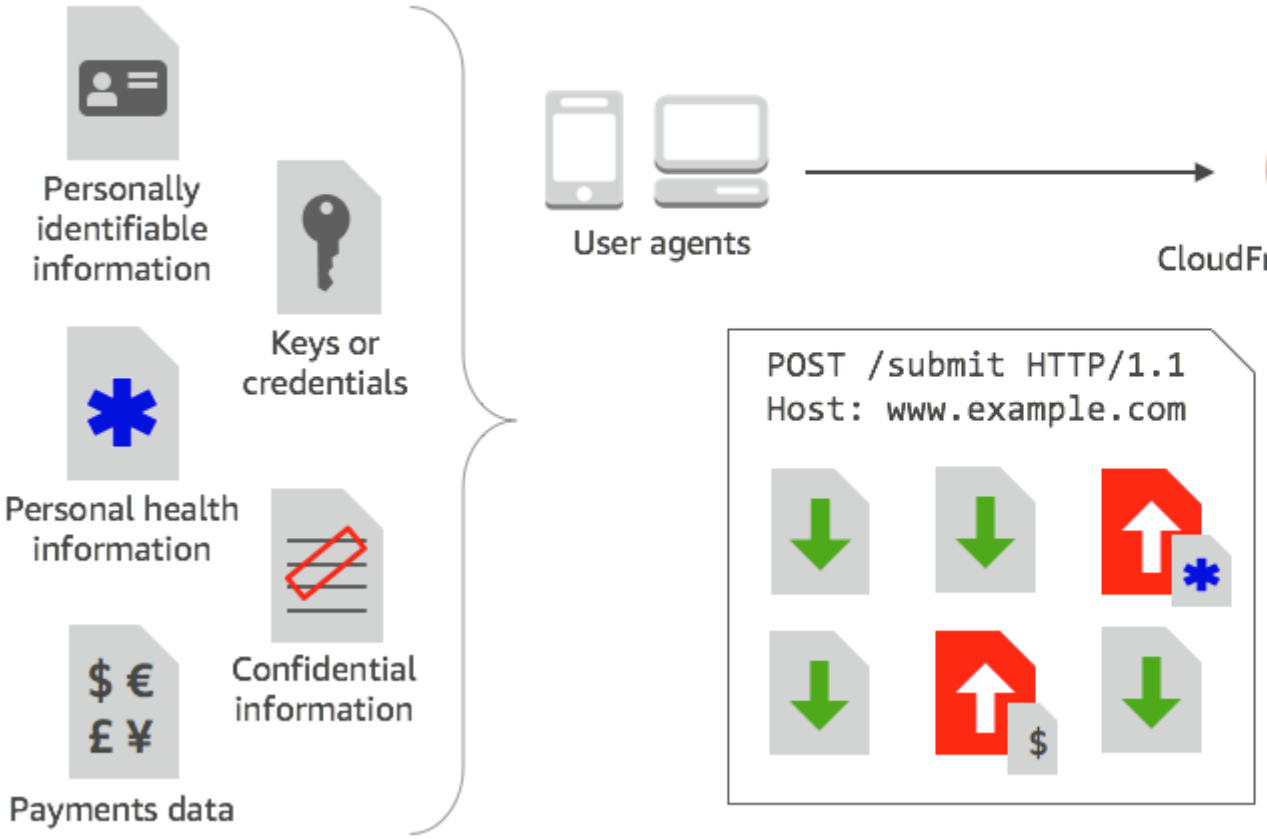
이미 CloudFront를 구성하여 HTTPS를 사용한 오리진 서버로의 보안 종단 간 연결 적용을 지원할 수 있습니다. 필드 레벨 암호화는 HTTPS와 함께 추가 보안 레이어를 추가하여 시스템 처리 전체에서 특정 데이터를 보호하고 특정 애플리케이션만 이를 볼 수 있도록 합니다.

필드 레벨 암호화를 통해 사용자가 제출한 민감한 정보를 웹 서버에 안전하게 업로드할 수 있습니다. 클라이언트가 제공한 민감한 정보는 사용자에게 더욱 가까운 엣지에서 암호화되고, 전체 애플리케이션 스택에서 암호화를 유지하여 데이터를 필요로 하고 이를 해독할 자격 증명을 보유한 애플리케이션만이 이를 가능하도록 할 수 있습니다.

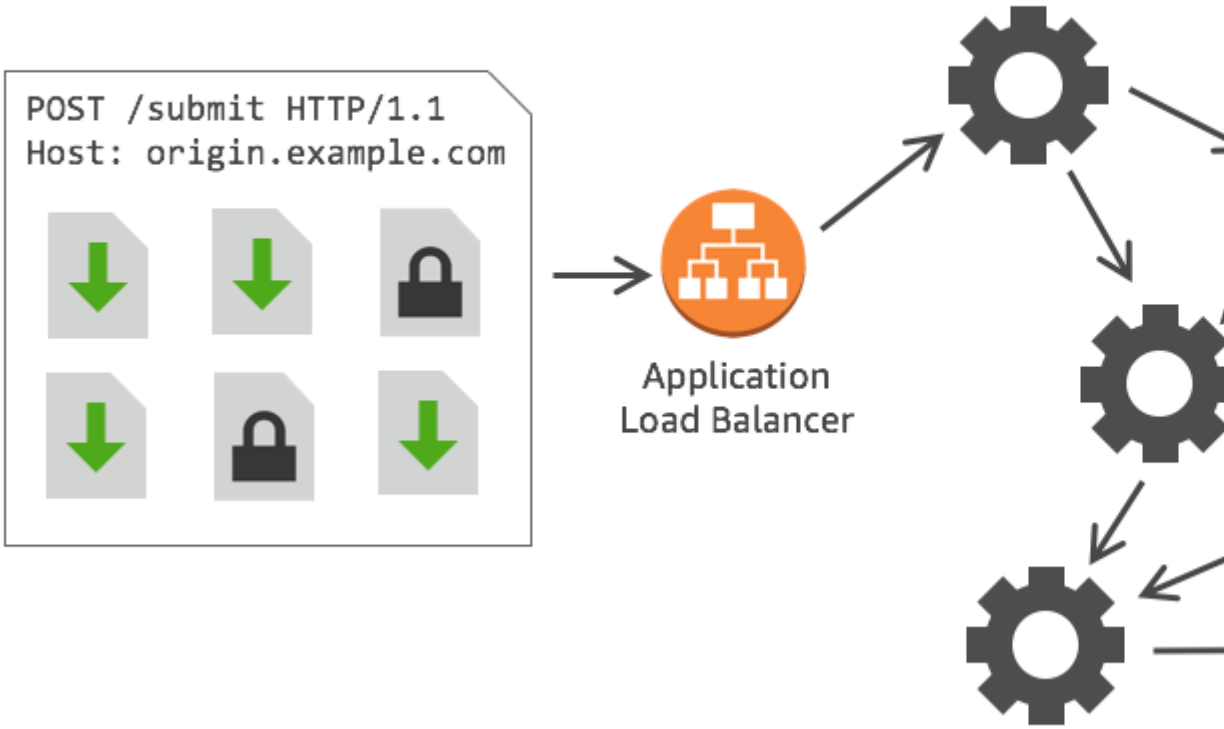
필드 레벨 암호화를 사용하려면 CloudFront 배포를 구성하여 암호화되기를 원하는 POST 요청의 필드 세트와 암호화에 사용할 퍼블릭 키를 지정할 수 있습니다. 한 요청에서 최대 10개의 데이터 필드를 암호화할 수 있습니다. (필드 레벨 암호화된 요청의 모든 데이터를 암호화할 수는 없습니다. 암호화할 개별 필드를 지정해야 합니다.)

필드 레벨 암호화가 포함된 HTTPS 요청이 오리진에 전달될 때 요청은 오리진 하위 시스템 전체에 라우팅되고, 민감한 데이터는 계속 암호화되어 민감한 데이터의 침해 또는 우발적 데이터 손실의 위험을 줄입니다. 크레딧 번호에 대한 액세스를 필요로 하는 결제 처리 시스템과 같이 업무상 사유에 따라 민감한 데이터에 대한 액세스를 필요로 하는 구성 요소는 적절한 프라이빗 키를 사용하여 해독하고 데이터에 액세스할 수 있습니다.

필드 레벨 암호화를 사용하려면 오리진이 chunked 인코딩을 지원해야 합니다.



CloudFront 필드 레벨 암호화는 퍼블릭 키 암호화라고도 하는 비대칭 암호화를 사용합니다. CloudFront에 퍼블릭 키를 입력하면 지정한 모든 민감한 데이터가 자동적으로 암호화됩니다. CloudFront에 입력한 키는 암호화된 값의 해독에 사용할 수 없습니다. 이는 프라이빗 키만이 가능합니다.



주제

- 필드 레벨 암호화 개요 (p. 187)
- 필드 레벨 암호화 설정 (p. 188)
- 오리진의 데이터 필드 해독 (p. 191)

필드 레벨 암호화 개요

다음 단계는 필드 레벨 암호화 설정의 개요를 제공합니다. 자세한 단계는 필드 레벨 암호화 설정 (p. 188) 단원을 참조하십시오.

1. 퍼블릭 키-프라이빗 키 페어를 가져옵니다. CloudFront에서 필드 레벨 암호화를 설정하기 전에 퍼블릭 키를 얻고 이를 추가해야 합니다.
2. 필드 레벨 암호화 프로필을 생성합니다. CloudFront에서 생성한 필드 레벨 암호화 프로필은 암호화하고자 하는 필드를 정의합니다.
3. 필드 레벨 암호화 구성을 생성합니다. 구성은 요청의 콘텐츠 유형 또는 쿼리 인수를 기반으로 특정 데이터 필드 암호화에 사용할 프로필을 지정합니다. 각기 다른 시나리오에 대해 원하는 요청 전달 동작 옵션을 선택할 수도 있습니다. 예를 들어, 요청 URL의 쿼리 인수에 의해 프로필 이름이 지정될 때 이는 CloudFront.에 존재하지 않습니다.
4. 캐시 동작에 연결합니다. 배포에 대한 캐시 동작의 구성을 연결하여 CloudFront가 데이터를 암호화할 시기를 지정합니다.

필드 레벨 암호화 설정

다음 단계를 따라 필드 레벨 암호화 사용을 시작합니다. 필드 레벨 암호화에 대한 할당량(이전에는 제한이라고 함)에 대한 자세한 내용은 할당량 (p. 447) 단원을 참조하십시오.

- 1단계: RSA 키 페어 생성 (p. 188)
- 2단계: CloudFront에 퍼블릭 키 추가 (p. 188)
- 3단계: 필드 레벨 암호화 프로필 생성 (p. 188)
- 4단계: 구성 생성 (p. 189)
- 5단계: 캐시 동작에 구성 추가 (p. 191)

1단계: RSA 키 페어 생성

시작하려면 CloudFront가 데이터를 암호화할 수 있도록 퍼블릭 키를 포함하는 RSA 키 페어를 만들어야 합니다. 또한 오리진의 구성 요소가 암호화된 필드를 해독할 수 있도록 프라이빗 키를 만들어야 합니다. OpenSSL 또는 기타 도구를 사용해서 키 페어를 만들 수 있습니다. 키 크기는 2048비트여야 합니다.

예를 들어 OpenSSL을 사용하는 경우, 다음 명령으로 길이가 2048비트인 키 페어를 만들고 `private_key.pem` 파일에 저장할 수 있습니다.

```
openssl genrsa -out private_key.pem 2048
```

이렇게 만들어진 파일은 퍼블릭 키와 프라이빗 키를 모두 포함합니다. 그 파일에서 퍼블릭 키를 추출하려면 다음 명령을 실행합니다.

```
openssl rsa -pubout -in private_key.pem -out public_key.pem
```

퍼블릭 키 파일(`public_key.pem`)에는 다음 단계에서 붙여 넣는 암호화된 키 값이 포함됩니다.

2단계: CloudFront에 퍼블릭 키 추가

RSA 키 페어를 얻은 후 퍼블릭 키를 CloudFront에 추가합니다.

퍼블릭 키를 CloudFront에 추가하려면(콘솔)

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 탐색 창에서 퍼블릭 키를 선택합니다.
3. Add public key(퍼블릭 키 추가)를 선택합니다.
4. 키 이름에 키의 고유 이름을 입력합니다. 이름은 공백을 포함할 수 없으며, 영숫자 문자, 밑줄(_) 및 하이픈(-)만 포함할 수 있습니다. 문자는 최대 128자입니다.
5. 키 값에서 " -----BEGIN PUBLIC KEY-----" 및 "-----END PUBLIC KEY-----" 행을 포함한 퍼블릭 키의 암호화된 키 값을 붙여 넣습니다.
6. 설명에 코멘트를 추가합니다(선택 사항). 예를 들어, 퍼블릭 키의 만료 날짜를 포함할 수 있습니다.
7. 추가를 선택합니다.

이 절차의 단계를 반복하여 CloudFront에서 사용할 키를 더 추가할 수 있습니다.

3단계: 필드 레벨 암호화 프로필 생성

CloudFront에 하나 이상의 퍼블릭 키를 추가한 이후 CloudFront에 어떤 필드를 암호화할지 알리는 프로필을 생성합니다.

필드 레벨 암호화 프로필을 생성하려면(콘솔)

1. 왼쪽 탐색 창에서 Field-level encryption(필드 레벨 암호화)을 선택합니다.
2. Create profile(프로파일 생성)을 선택합니다.
3. 다음 필드를 입력합니다.

   프로필 이름

   프로필의 고유한 이름을 입력합니다. 이름은 공백을 포함할 수 없으며, 영숫자 문자, 밑줄(_) 및 하이픈(-)만 포함할 수 있습니다. 문자는 최대 128자입니다.

   퍼블릭 키 이름

   2단계에서 CloudFront에 추가한 퍼블릭 키의 이름을 드롭다운 목록에서 선택합니다. CloudFront는 키를 사용하여 이 프로필에서 지정한 필드를 암호화합니다.

   공급자 이름

   키 페어를 얻은 공급자와 같이 키를 식별하는 데 도움이 될 구문을 입력합니다. 프라이빗 키와 함께 이 정보는 애플리케이션이 데이터 필드를 해독할 때 필요합니다. 공급자 이름은 공백을 포함할 수 없으며, 영숫자 문자, 콜론(:), 밑줄(_) 및 하이픈(-)만 포함할 수 있습니다. 문자는 최대 128자입니다.

   일치시킬 필드 이름 패턴

   CloudFront에서 암호화하고자 하는 데이터 필드의 이름 또는 요청에서 데이터 필드 이름을 식별하는 패턴을 입력합니다. + 옵션을 사용하여 이 키를 사용하여 암호화하고자 하는 모든 필드를 추가합니다.

   필드 이름 패턴의 경우 데이터 필드의 전체 이름(예: DateOfBirth)을 입력하거나 와일드카드 문자(*)를 포함하여 이름의 첫 부분(예: CreditCard*)만을 입력할 수 있습니다. 필드 이름 패턴은 영숫자 문자, 대괄호([및]), 마침표(.), 밑줄(_) 및 하이픈(-)만을 포함할 수 있으며, 추가로 와일드카드 문자(*)를 선택적으로 포함할 수 있습니다.

   다른 필드 이름 패턴과 중첩되는 문자를 사용하지 마세요. 예를 들어, ABC*라는 필드 이름 패턴을 보유한 경우 또 다른 AB* 필드 이름 패턴을 추가할 수는 없습니다. 추가로 필드 이름은 대소문자를 구분하며, 사용할 수 있는 최대 문자는 128자입니다.

   Comment

   (선택 사항) 이 프로필에 대한 설명을 입력합니다. 사용할 수 있는 최대 문자 수는 128자입니다.
4. 필드에 내용을 입력한 후 Create profile(프로파일 생성)을 선택합니다.
5. 프로필을 더 추가하고자 하는 경우 프로필 추가를 선택합니다.

4단계: 구성 생성

하나 이상의 필드 레벨 암호화 프로필을 생성한 이후, 암호화할 데이터를 포함하는 요청의 콘텐츠 유형, 암호화에 사용할 프로필, CloudFront의 암호화 처리 방법을 지정하는 기타 옵션을 지정하는 구성을 생성합니다.

예를 들어, CloudFront가 데이터를 암호화할 수 없을 때 다음 시나리오에서 CloudFront가 요청을 차단하거나 오리진으로 전달할지 여부를 지정할 수 있습니다.

- 요청의 콘텐츠 유형이 구성에 없을 때. 구성에 콘텐츠 유형을 추가하지 않은 경우 CloudFront가 해당 콘텐츠 유형을 포함한 요청을 데이터 필드 암호화 없이 오리진에 전달할지 또는 요청을 차단하고 오류를 반환할지 여부를 지정할 수 있습니다.

  참고: 구성에 콘텐츠 유형을 추가했지만 해당 유형을 사용할 프로필을 지정하지 않은 경우 해당 콘텐츠 유형을 포함한 요청은 항상 오리진으로 전달됩니다.
- 쿼리 인수에 입력된 프로필 이름이 알 수 없는 이름임. 배포에 존재하지 않는 프로필 이름을 포함하는 `fle-profile` 쿼리 인수를 지정할 때 CloudFront가 요청을 데이터 필드 암호화 없이 오리진으로 보낼지 또는 요청을 차단하고 오류를 반환할지 여부를 지정할 수 있습니다.

구성에서 URL에 쿼리 인수로 프로필을 입력함으로써 해당 쿼리에 대한 콘텐츠 유형으로 매핑한 프로필을 무시할지 여부를 지정할 수도 있습니다. 기본적으로 CloudFront는 콘텐츠 유형에 매핑한 프로필을 사용합니다 (이를 지정한 경우). 이를 통해 기본적으로 사용할 프로필을 가질 수 있지만 다른 프로필을 적용하고자 하는 특정 요청을 정할 수도 있습니다.

예를 들어, 사용할 쿼리 인수로 (구성에 있는) SampleProfile을 지정할 수 있습니다. 그런 다음, `https://d1234.cloudfront.net` 대신 URL `https://d1234.cloudfront.net?fle-profile=SampleProfile`을 사용하여 CloudFront가 요청의 콘텐츠 유형에 대해 설정한 프로필 대신 이 요청에서 SampleProfile을 사용하도록 할 수 있습니다.

단일 계정에 대해 최대 10개의 구성을 생성할 수 있으며 계정에 대한 모든 배포의 캐시 동작에 구성 중 하나를 연결할 수 있습니다.

필드 레벨 암호화 구성을 생성하려면(콘솔)

1. Field-level encryption(필드 레벨 암호화) 페이지에서 구성 만들기를 선택합니다.

   참고: 최소 하나의 프로필을 생성하지 않은 경우 구성 생성 옵션이 표시되지 않습니다.
2. 다음 필드를 입력하여 사용할 프로필을 지정합니다. (일부 필드는 변경할 수 없습니다.)

   콘텐츠 유형(변경할 수 없음)

   콘텐츠 유형은 `application/x-www-form-urlencoded`로 설정되며 변경할 수 없습니다.

   기본 프로필 ID(선택 사항)

   드롭다운 목록 가운데 콘텐츠 유형 필드에서 콘텐츠 유형을 매핑하고자 하는 프로필을 선택합니다.

   콘텐츠 형식(변경할 수 없음)

   콘텐츠 형식은 `URLencoded`로 설정되며 변경할 수 없습니다.
3. 다음 옵션에 대한 CloudFront 기본 동작을 변경하고자 하는 경우 해당 확인란을 선택합니다.

   요청의 콘텐츠 유형이 구성되지 않을 때 오리진으로 요청 전달

   요청의 콘텐츠 유형에 대해 사용할 프로필을 지정하지 않았다면 요청이 오리진으로 이동하도록 허용할 경우 확인란을 선택합니다.

   입력된 쿼리 인수를 포함한 콘텐츠 유형에 대한 프로필 무시

   쿼리 인수에 입력된 프로필이 콘텐츠 유형에 대해 지정한 프로필을 무시하도록 허용할 경우 확인란을 선택합니다.
4. 확인란을 선택하여 쿼리 인수가 기본 프로필을 무시하도록 허용할 경우 구성에 대한 다음 추가 필드를 입력해야 합니다. 쿼리에 사용할 쿼리 인수 매핑 가운데 최대 5개를 생성할 수 있습니다.

   쿼리 인수

   `fle-profile` 쿼리 인수에 대해 URL에서 포함하고자 하는 값을 입력합니다. 이 값은 CloudFront에 이 쿼리의 필드 레벨 암호화에 대한 쿼리 인수와 연결된 프로필 ID(다음 필드에서 지정)를 사용하도록 합니다.

   사용할 수 있는 최대 문자 수는 128자입니다. 값에는 공백이 포함될 수 없으며, 영숫자 문자와 다음 문자만을 사용할 수 있습니다. 대시(-), 마침표(.), 밑줄(_), 별표(*) 더하기 기호(+), 퍼센트(%).

   프로필 ID

   드롭다운 목록 가운데 쿼리 인수에 입력한 값과 연결하고자 하는 프로필을 선택합니다.

   쿼리 인수에 지정된 프로필이 존재하지 않을 때 요청을 오리진에 전달

   쿼리 인수에서 지정된 프로필이 CloudFront에서 정의되지 않은 경우 요청이 오리진으로 이동하도록 허용하려면 이 확인란을 선택합니다.

5단계: 캐시 동작에 구성 추가

필드 레벨 암호화를 사용하려면 배포에 대한 값으로 구성 ID를 추가함으로써 배포에 대한 캐시 동작과 구성을 연결합니다. 캐시 동작과 구성을 연결하려면 Viewer Protocol Policy와 Origin Protocol Policy는 HTTPS여야 합니다.

자세한 내용은 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.

오리진의 데이터 필드 해독

CloudFront는 AWS 암호화 SDK를 사용하여 데이터 필드를 암호화합니다. 데이터는 애플리케이션 스택에 걸쳐 암호화를 유지하며, 이를 해독할 자격 증명을 보유한 애플리케이션만이 액세스할 수 있습니다.

암호화 이후 암호 텍스트는 base64 암호화됩니다. 애플리케이션이 오리진의 텍스트를 해독할 때 우선 암호 텍스트를 디코딩한 다음 AWS 암호화 SDK를 사용하여 데이터를 해독해야 합니다.

다음 코드 샘플은 애플리케이션이 오리진의 데이터를 해독하는 방법을 보여줍니다. 다음을 참조하십시오.

- 예제를 간소화하기 위해 이 샘플은 작업 디렉터리에 있는 파일에서 퍼블릭 및 프라이빗 키(DER 형식)를 로드합니다. 실제로는 프라이빗 키를 오프라인 하드웨어 보안 모듈과 같은 안전한 오프라인 위치에 저장하고, 퍼블릭 키를 개발 팀에 배포합니다.
- CloudFront는 데이터를 암호화하는 동안 특정 정보를 사용하며 동일한 파라미터 세트를 오리진에서 사용하여 이를 해독해야 합니다. MasterKey를 초기화하는 동안 CloudFront가 사용하는 파라미터는 다음을 포함합니다.
  - PROVIDER_NAME: 필드 레벨 암호화 프로필을 생성할 때 이 값을 지정했습니다. 여기에서 동일한 값을 사용합니다.
  - KEY_NAME: 퍼블릭 키를 CloudFront로 업로드할 때 퍼블릭 키의 이름을 생성했으며, 프로필에서 키 이름을 지정했습니다. 여기에서 동일한 값을 사용합니다.
  - ALGORITHM: CloudFront는 "RSA/ECB/OAEPWithSHA-256AndMGF1Padding"을 암호화 알고리즘으로 사용합니다. 따라서 동일한 알고리즘을 사용하여 데이터를 해독해야 합니다.
- 입력으로 암호 텍스트를 포함한 다음 샘플 프로그램을 실행하는 경우 암호화된 데이터가 콘솔에 출력됩니다. 자세한 내용은 AWS 암호화 SDK의 Java 예제 코드를 참조하십시오.

샘플 코드

```
import java.nio.file.Files;
import java.nio.file.Paths;
import java.security.KeyFactory;
import java.security.PrivateKey;
import java.security.PublicKey;
import java.security.spec.PKCS8EncodedKeySpec;
import java.security.spec.X509EncodedKeySpec;

import org.apache.commons.codec.binary.Base64;

import com.amazonaws.encryptionsdk.AwsCrypto;
import com.amazonaws.encryptionsdk.CryptoResult;
import com.amazonaws.encryptionsdk.jce.JceMasterKey;

/**
 * Sample example of decrypting data that has been encrypted by CloudFront Field-Level
 Encryption.
 */
public class DecryptExample {

    private static final String PRIVATE_KEY_FILENAME = "private_key.der";
```

```
    private static final String PUBLIC_KEY_FILENAME = "public_key.der";
    private static PublicKey publicKey;
    private static PrivateKey privateKey;

    // CloudFront uses the following values to encrypt data, and your origin must use same
 values to decrypt it.
    // In your own code, for PROVIDER_NAME, use the provider name that you specified when
 you created your Field Level
    // Encryption Profile. This sample uses 'DEMO' for the value.
    private static final String PROVIDER_NAME = "DEMO";
    // In your own code, use the Key name that you specified when you added your public key
 to CloudFront. This sample
    // uses 'DEMOKEY' for the Key name.
    private static final String KEY_NAME = "DEMOKEY";
    // Cloudfront uses this algorithm when encrypting data.
    private static final String ALGORITHM = "RSA/ECB/OAEPWithSHA-256AndMGF1Padding";

    public static void main(final String[] args) throws Exception {

        final String dataToDecrypt = args[0];

        // This sample uses files to get public and private keys.
        // In practice, you should distribute the public key and save the private key in
 secure storage.
        populateKeyPair();

        System.out.println(decrypt(debase64(dataToDecrypt)));
    }

    private static String decrypt(final byte[] bytesToDecrypt) throws Exception {
        // You can decrypt the stream only by using the private key.

        // 1. Instantiate the SDK
        final AwsCrypto crypto = new AwsCrypto();

        // 2. Instantiate a JCE master key
        final JceMasterKey masterKey = JceMasterKey.getInstance(
                publicKey,
                privateKey,
                PROVIDER_NAME,
                KEY_NAME,
                ALGORITHM);

        // 3. Decrypt the data
        final CryptoResult <byte[], ? > result = crypto.decryptData(masterKey,
 bytesToDecrypt);
        return new String(result.getResult());
    }

    // Function to decode base64 cipher text.
    private static byte[] debase64(final String value) {
        return Base64.decodeBase64(value.getBytes());
    }

    private static void populateKeyPair() throws Exception {
        final byte[] PublicKeyBytes = Files.readAllBytes(Paths.get(PUBLIC_KEY_FILENAME));
        final byte[] privateKeyBytes = Files.readAllBytes(Paths.get(PRIVATE_KEY_FILENAME));
        publicKey = KeyFactory.getInstance("RSA").generatePublic(new
 X509EncodedKeySpec(PublicKeyBytes));
        privateKey = KeyFactory.getInstance("RSA").generatePrivate(new
 PKCS8EncodedKeySpec(privateKeyBytes));
    }
}
```

콘텐츠 캐싱 및 가용성 최적화

이 단원에서는 객체의 캐싱을 설정하고 관리하여 성능을 개선하고 비즈니스 요구 사항을 충족하는 방법을 설명합니다.

CloudFront가 제공하게 하려는 콘텐츠를 추가하고 제거하는 방법을 알아보려면 CloudFront가 배포하는 콘텐츠 추가, 제거 또는 바꾸기 (p. 77) 단원을 참조하십시오.

주제

- 캐싱과 CloudFront 엣지 캐시의 작동 방식 (p. 193)
- CloudFront 엣지 캐시에서 제공되는 요청의 비율 증가(캐시 적중률) (p. 193)
- 쿼리 문자열 파라미터 기반의 콘텐츠 캐싱 (p. 196)
- 쿠키를 기반으로 콘텐츠 캐싱 (p. 198)
- 요청 헤더에 따라 콘텐츠 캐싱 (p. 200)
- 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204)
- CloudFront 오리진 장애 조치를 통한 고가용성 최적화 (p. 209)
- CloudFront에서 객체에 대한 부분적인 요청을 처리하는 방법(Range GET) (p. 214)
- 기본 루트 객체 지정 (p. 215)

캐싱과 CloudFront 엣지 캐시의 작동 방식

CloudFront를 사용하는 목적 중 하나는 오리진 서버에서 직접 응답해야 하는 요청의 수를 줄이기 위한 것입니다. 이렇게 하면 사용자에게 보다 가까운 CloudFront 엣지 로케이션에서 더 많은 객체가 제공되므로 오리진 서버의 부하를 줄이고 지연 시간을 단축할 수 있습니다.

전체 요청 대비 CloudFront에서 엣지 캐시로부터 제공하는 요청의 비율이 더 클수록(즉, 캐시 적중률이 높을수록) 최신 버전 또는 고유한 버전의 객체를 가져오기 위해 CloudFront에서 오리진에 전달해야 할 최종 사용자 요청의 수는 줄어듭니다.

CloudFront 콘솔에서 최종 사용자 요청의 적중, 누락, 오류 백분율을 확인할 수 있습니다. 자세한 내용은 CloudFront 캐시 통계 보고서 (p. 371) 단원을 참조하십시오.

적중률에 영향을 주는 요인은 매우 많습니다. CloudFront 엣지 캐시에서 제공되는 요청의 비율 증가(캐시 적중률) (p. 193)의 지침을 수행하여 CloudFront 배포 구성을 조정하고 적중률을 높일 수 있습니다.

CloudFront 엣지 캐시에서 제공되는 요청의 비율 증가(캐시 적중률)

오리진 서버에서 콘텐츠를 제공하는 대신 CloudFront 엣지 캐시에서 콘텐츠를 제공하는 최종 사용자 요청을 비율을 높임으로써, 즉 배포에 대한 캐시 적중률을 높여서 성능을 향상시킬 수 있습니다.

다음 섹션에서는 캐시 적중률을 개선하는 방법을 설명합니다.

주제

- CloudFront에서 객체를 캐싱하는 시간 지정 (p. 194)
- 쿼리 문자열 파라미터 기반 캐싱 (p. 194)
- 쿠키 값 기반 캐싱 (p. 194)
- 요청 헤더 기반 캐싱 (p. 195)

- 압축이 불필요할 때 Accept-Encoding 헤더 제거 (p. 195)
- HTTP를 사용한 미디어 콘텐츠 제공 (p. 196)

CloudFront에서 객체를 캐싱하는 시간 지정

캐시 적중률을 높이기 위해 객체에 `Cache-Control max-age` 명령을 추가하도록 오리진을 구성하고 `max-age`에 실질적으로 가장 긴 값을 지정할 수 있습니다. 캐시 주기가 짧을수록 CloudFront에서는 또 다른 요청을 오리진에 더 자주 전달하여 객체가 변경되었는지 여부를 확인하고 변경된 경우 최신 버전을 가져옵니다. 자세한 내용은 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204) 단원을 참조하십시오.

쿼리 문자열 파라미터 기반 캐싱

쿼리 문자열 파라미터를 기반으로 캐싱하도록 CloudFront를 구성한 경우, 다음을 수행하면 캐싱 성능을 개선할 수 있습니다.

- 오리진에서 고유한 객체를 반환하는 쿼리 문자열 파라미터만 전달하도록 CloudFront를 구성합니다.
- 동일 파라미터의 전체 인스턴스에 대해서는 대문자 또는 소문자만 사용합니다. 예를 들어, 요청 하나에 `parameter1=A`가 포함되어 있고, 다른 요청에는 `parameter1=a`가 포함된 경우 CloudFront에서는 요청에 각각 `parameter1=A`와 `parameter1=a`가 포함되어 있을 때 오리진으로 개별 요청을 전달합니다. 그런 다음 CloudFront에서는 오리진에서 개별적으로 반환한 객체를 각각 캐싱합니다. 이는 이 객체들이 실제로는 동일하더라도 마찬가지입니다. `A` 또는 `a`만 사용하는 경우, CloudFront에서는 오리진에 더 적은 요청을 전달하게 됩니다.
- 파라미터를 같은 순서로 나열합니다. 다른 순서로 나열하는 경우, 객체에 대한 하나의 요청에 쿼리 문자열 `parameter1=a&parameter2=b`가 포함되어 있고 같은 객체의 다른 요청에 가 포함되어 있다면 `parameter2=b&parameter1=a`, CloudFront에서는 두 요청을 오리진에 모두 전달하고 해당 객체가 실제로는 동일한 경우라도 이를 개별적으로 캐싱합니다. 파라미터에 언제나 같은 순서를 사용한다면 CloudFront에서는 오리진에 더 적은 요청을 전달하게 됩니다.

자세한 내용은 쿼리 문자열 파라미터 기반의 콘텐츠 캐싱 (p. 196) 단원을 참조하십시오. CloudFront에서 오리진에 전달한 쿼리 문자열을 검토하려는 경우, CloudFront 액세스 로그를 활성화하고 로그 파일에서 `cs-uri-query` 열의 값을 확인합니다. 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

쿠키 값 기반 캐싱

쿠키 값을 기반으로 캐싱하도록 CloudFront를 구성한 경우, 다음을 수행하면 캐싱 성능을 개선할 수 있습니다.

- 전체 쿠키를 전달하는 대신 지정된 쿠키만 전달하도록 CloudFront를 구성합니다. CloudFront에서 오리진에 전달하도록 구성한 쿠키에 대해서는, CloudFront에서 쿠키 이름과 값의 모든 조합을 전달하고 오리진에서 반환하는 객체를 개별적으로 캐싱합니다. 이는 해당 객체가 실제로는 동일한 경우라도 마찬가지입니다.

  예를 들어, 최종 사용자가 각 요청에 두 개의 쿠키를 포함했다고 가정하면, 각 쿠키는 3가지 값을 가질 수 있으며 이 모든 쿠키 값의 조합이 가능합니다. CloudFront에서는 각 객체에 대해 오리진에 최대 6가지 요청을 전달합니다. 오리진에서 오직 하나의 쿠키를 기반으로 서로 다른 버전의 객체를 반환하는 경우, CloudFront에서는 오리진에 필요한 것 이상의 요청을 전달하게 되며 불필요하게 동일한 버전의 객체를 여러 번 캐싱하게 됩니다.
- 정적 콘텐츠와 동적 콘텐츠의 캐시 동작을 별도로 만들고 오리진에는 동적 콘텐츠에 대해서만 쿠키를 전달하도록 CloudFront를 구성합니다.

  예를 들어, 배포에 대한 하나의 캐시 동작이 있고 .js 파일 및 .css 파일과 같은 거의 변동이 없는 두 동적 콘텐츠에 대한 배포를 사용 중이라고 가정합니다. CloudFront에서는 쿠키 값을 바탕으로 .css 파일의 개별 버전을 캐싱하여 CloudFront 엣지 로케이션에서 각각의 새 쿠키 값 또는 쿠키 값들의 조합에 대한 요청을 오리진에 전달하도록 합니다.

경로 패턴이 *.css이고 CloudFront에서 쿠키 값에 따라 캐싱하지 않는 캐시 동작을 만든 경우, CloudFront에서는 .css 파일에 대한 요청 중 엣지 로케이션에서 수신하는 지정된 .css 파일에 대한 첫 번째 요청과 .css 파일이 만료된 후의 첫 번째 요청만 오리진에 전달합니다.

- 가능한 경우, 각 사용자(예: 사용자 ID)별로 쿠키 값이 고유한 동적 콘텐츠에 대해서와 더 작은 수의 고유 값에 따라 달라지는 동적 콘텐츠에 대해서 별도의 캐시 동작을 만듭니다

자세한 내용은 쿠키를 기반으로 콘텐츠 캐싱 (p. 198) 단원을 참조하십시오. CloudFront에서 오리진에 전달한 쿠키를 검토하려는 경우, CloudFront 액세스 로그를 활성화하고 로그 파일에서 `cs(Cookie)` 열의 값을 확인합니다. 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

요청 헤더 기반 캐싱

요청 헤더를 기반으로 캐싱하도록 CloudFront를 구성한 경우, 다음을 수행하면 캐싱 성능을 개선할 수 있습니다.

- 전체 헤더를 기반으로 전달 및 캐싱하는 대신 지정된 헤더를 기반으로만 전달하고 캐싱하도록 CloudFront를 구성합니다. CloudFront에서는 지정한 헤더에 대해 헤더 이름과 값의 모든 조합을 전달하고 오리진에서 반환하는 객체를 개별적으로 캐싱합니다. 이는 해당 객체가 실제로는 동일한 경우라도 마찬가지입니다.

  Note

  CloudFront에서는 다음 주제에서 지정한 헤더를 오리진에 항상 전달합니다.

  - CloudFront에서 요청을 처리하고 Amazon S3 오리진 서버에 요청을 전달하는 방법 > CloudFront에서 제거하거나 업데이트하는 HTTP 요청 헤더 (p. 233)
  - CloudFront에서 요청을 처리하고 사용자 지정 오리진 서버에 요청을 전달하는 방법 > HTTP 요청 헤더 및 CloudFront 동작(사용자 지정 및 S3 오리진) (p. 239)

  요청 헤더에 따라 캐싱하도록 CloudFront를 구성한 경우, CloudFront에서 헤더 값에 따라 객체를 캐싱하는 경우를 제외하고는 CloudFront에서 전달하는 헤더를 변경하지 않습니다.
- 큰 수의 고유 값을 갖는 요청 헤더를 기반으로 캐싱하지 않도록 하십시오.

  예를 들어 사용자 디바이스에 따라 서로 다른 크기의 이미지를 제공하려는 경우, 매우 큰 값을 가질 수 있는 `User-Agent` 헤더를 기반으로 캐싱하도록 CloudFront를 구성하지 마십시오. 대신 CloudFront 디바이스 유형 헤더인 `CloudFront-Is-Desktop-Viewer`, `CloudFront-Is-Mobile-Viewer`, `CloudFront-Is-SmartTV-Viewer`, `CloudFront-Is-Tablet-Viewer`를 기반으로 캐싱하도록 CloudFront를 구성하십시오. 또한 태블릿과 데스크톱에 대해 같은 버전의 이미지를 반환하려는 경우, `CloudFront-Is-Tablet-Viewer` 헤더가 아닌 `CloudFront-Is-Desktop-Viewer` 헤더만 전달하세요.

자세한 내용은 요청 헤더에 따라 콘텐츠 캐싱 (p. 200) 단원을 참조하십시오.

압축이 불필요할 때 Accept-Encoding 헤더 제거

기본적으로 CloudFront에서는 요청 수신 시 `Accept-Encoding` 헤더의 값을 확인합니다. 헤더의 값에 `gzip`이 포함되는 경우에는 CloudFront에서 헤더와 값의 `gzip`—`Accept-Encoding: gzip`—을 캐시 키에 추가한 후 오리진으로 전달합니다. 이 동작을 통해 CloudFront에서는 객체나 해당 객체의 압축 버전을 `Accept-Encoding` 헤더의 값에 따라 제공할 수 있습니다.

압축이 오리진이나 CloudFront에서 지원되지 않거나 압축 자체가 불가한 콘텐츠라서—압축 활성화가 어렵다면—다른 동작을 지정하여 캐시 적중률을 높일 수 있습니다. 이렇게 하려면 배포의 캐시 동작을 다음과 같이 Custom Origin Header를 설정하는 오리진에 연결하십시오.

- Header name(헤더 이름): `Accept-Encoding`
- 헤더 값: (공란으로 둘 것)

이 구성을 사용할 때는 CloudFront에서 Accept-Encoding 헤더를 오리진으로 전달하는 일 없이 캐시 키에서 제거합니다. 이 구성을 지정하면 CloudFront가 해당 오리진에서의 배포를 통해 제공하는 모든 콘텐츠에 적용된다는 점에 유의하십시오.

HTTP를 사용한 미디어 콘텐츠 제공

온디맨드 비디오(VOD) 및 스트리밍 비디오 콘텐츠를 최적화하는 방법에 대한 자세한 내용은 CloudFront를 사용한 온디맨드 비디오 및 라이브 스트리밍 비디오 (p. 261) 단원을 참조하십시오.

쿼리 문자열 파라미터 기반의 콘텐츠 캐싱

일부 웹 애플리케이션은 쿼리 문자열을 사용하여 오리진에 정보를 전송합니다. 쿼리 문자열은 ? 문자 다음에 오는 웹 요청의 일부이며 이 문자열은 & 문자로 구분된 하나 이상의 파라미터를 포함할 수 있습니다. 다음 예의 쿼리 문자열에는 *color=red*와 *size=large*의 두 가지 파라미터가 포함되어 있습니다.

`http://d111111abcdef8.cloudfront.net/images/image.jpg?`*`color=red&size=large`*

웹 배포의 경우 CloudFront에서 쿼리 문자열을 오리진에 전달하도록 선택할 수 있으며, 그럴 경우 콘텐츠를 모든 파라미터를 기준으로 캐싱할지 아니면 일부 파라미터를 기준으로 캐싱할지를 선택합니다. 이 기능이 유용한 이유를 알아보려면 다음 예제를 살펴보십시오.

웹사이트가 5개 언어로 제공되는 경우를 가정하겠습니다. 5개 웹사이트 버전의 디렉터리 구조와 파일 이름은 동일합니다. 사용자가 웹사이트를 볼 때 CloudFront로 전달되는 요청에는 사용자가 선택한 언어를 기준으로 하는 언어 쿼리 문자열 파라미터가 포함됩니다. 쿼리 문자열을 오리진으로 전달하고 언어 파라미터를 기준으로 캐싱하도록 CloudFront를 구성할 수 있습니다. 웹 서버에서 선택한 언어에 해당하는 특정 페이지의 버전을 반환하도록 구성할 경우 CloudFront는 언어 쿼리 문자열 파라미터 값을 기준으로 각 언어 버전을 별도로 캐싱합니다.

이 예에서 웹 사이트의 메인 페이지가 main.html인 경우 다음과 같은 5개 요청이 발생하면 CloudFront에서 언어 쿼리 문자열 파라미터의 각 값에 대해 한 번씩 main.html을 다섯 번 캐싱합니다.

- `http://d111111abcdef8.cloudfront.net/main.html?`*`language=de`*
- `http://d111111abcdef8.cloudfront.net/main.html?`*`language=en`*
- `http://d111111abcdef8.cloudfront.net/main.html?`*`language=es`*
- `http://d111111abcdef8.cloudfront.net/main.html?`*`language=fr`*
- `http://d111111abcdef8.cloudfront.net/main.html?`*`language=jp`*

다음을 참조하십시오.

- RTMP 배포의 경우 CloudFront에서 쿼리 문자열 파라미터를 오리진에 전달하도록 구성할 수 없습니다. RTMP 배포를 사용하는 경우, CloudFront는 오리진 서버로 요청을 전달하기 전에 모든 쿼리 문자열 파라미터를 제거합니다.
- 일부 HTTP 서버는 쿼리 문자열 파라미터를 처리하지 않으므로 파라미터 값을 기준으로 객체의 다른 버전을 반환하지 않습니다. 이러한 오리진의 경우 CloudFront에서 쿼리 문자열 파라미터를 오리진으로 전달하도록 구성할 경우 CloudFront는 오리진에서 모든 파라미터 값에 대해 CloudFront에 동일한 객체 버전을 반환할 경우에도 파라미터 값을 기준으로 캐싱합니다.
- 쿼리 문자열 파라미터가 위의 예제에서 설명한 대로 작동하려면 쿼리 문자열 매개 변수 간에 "&"문자를 구분 기호로 사용해야 합니다. 다른 구분 기호를 사용하는 경우에는 캐싱 기준으로 CloudFront에서 지정하는 파라미터 및 쿼리 문자열에서 파라미터가 나타나는 순서에 따라 결과가 달라질 수 있습니다.

  다음 예는 다른 구분 기호를 사용하고, `color` 파라미터에 따라서만 캐싱하도록 CloudFront를 구성할 경우의 결과를 보여줍니다.

  - 다음 요청에서 CloudFront는 `color` 파라미터 값을 기준으로 콘텐츠를 캐싱하지만 CloudFront는 이 값을 *`red;size=large`*로 해석합니다.

```
http://d111111abcdef8.cloudfront.net/images/image.jpg?color=red;size=large
```

- 다음 요청에서 CloudFront는 콘텐츠를 캐싱하지만 쿼리 문자열 파라미터를 기준으로 캐싱하지 않습니다. 그 이유는 CloudFront에서 `color` 파라미터를 기준으로 캐싱하도록 구성했지만 CloudFront는 다음 문자열에 *`large;color=red`* 값을 가진 `size` 파라미터만 포함된 것으로 해석하기 때문입니다.

```
http://d111111abcdef8.cloudfront.net/images/image.jpg?size=large;color=red
```

CloudFront에서 다음 중 하나를 수행하도록 구성할 수 있습니다.

- 쿼리 문자열을 오리진에 전혀 전달하지 않음. 쿼리 문자열을 전달하지 않을 경우 CloudFront는 쿼리 문자열 파라미터를 기준으로 캐싱하지 않습니다.
- 쿼리 문자열을 오리진으로 전달하며 쿼리 문자열의 모든 파라미터를 기준으로 캐싱함
- 쿼리 문자열을 오리진으로 전달하며 쿼리 문자열 중 지정된 파라미터를 기준으로 캐싱함

자세한 내용은 캐싱 최적화 (p. 197) 단원을 참조하십시오.

주제

- 쿼리 문자열 전달 및 캐싱에 대한 콘솔 및 API 설정 (p. 197)
- 캐싱 최적화 (p. 197)
- 쿼리 문자열 파라미터 및 CloudFront 액세스 로그 (p. 198)

쿼리 문자열 전달 및 캐싱에 대한 콘솔 및 API 설정

CloudFront 콘솔에서 쿼리 문자열 전달 및 캐싱을 구성하려면 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35)에서 다음 설정을 참조하십시오.

- Query String Forwarding and Caching (p. 49)
- Query String Whitelist (p. 49)

CloudFront API를 사용하여 쿼리 문자열 전달 및 캐싱을 구성하려면 Amazon CloudFront API Reference의 DistributionConfig 복합 유형 및 DistributionConfigWithTags 복합 유형에서 다음 설정을 참조하십시오.

- `QueryString`
- `QueryStringCacheKeys`

캐싱 최적화

CloudFront에서 쿼리 문자열 파라미터를 기준으로 캐싱하도록 구성할 경우 CloudFront에서 오리진에 전달하는 요청 수를 줄이는 방법은 다음과 같습니다. 이 요청 수를 줄이면 CloudFront 엣지 로케이션에서 더 많은 객체를 제공하므로 오리진 서버의 부하가 감소하며 지연 시간도 감소합니다.

오리진에서 다른 객체 버전을 반환하는 경우 파라미터만 기준으로 캐싱

웹 애플리케이션에서 CloudFront로 각 쿼리 문자열 파라미터를 전달할 때마다 CloudFront는 모든 파라미터 값에 대해 오리진에 요청을 전달하며 모든 파라미터 값에 대해 별도의 객체 버전을 캐싱합니다. 이는 오리진에서 파라미터 값과 상관없이 항상 같은 객체를 반환하는 경우라도 그렇습니다. 복수 파라미터의 경우 요청 수와 객체 수를 곱합니다. 객체에 대한 요청에 2개의 파라미터가 있고 각각 세 가지 값이 있는 경우 CloudFront는 해당 객체의 6개 버전을 캐싱합니다(이 섹션의 다른 권장 사항을 따르는 경우).

오리진에서 다른 버전을 반환하는 쿼리 문자열 파라미터만 기준으로 캐싱하도록 CloudFront를 구성하고 각 파라미터를 기준으로 캐싱하는 이점을 주의 깊게 고려하는 것이 좋습니다. 예를 들어, 소매 웹사이

트가 있다고 가정해봅시다. 6가지 색상의 자켓 사진이 있고 이 자켓은 10가지 사이즈로 제공됩니다. 가지고 있는 자켓 사진은 다른 색상을 보여주지만 다른 사이즈는 보여주지 않습니다. 캐싱을 최적화하려면 CloudFront에서 사이즈 파라미터가 아닌 색상 파라미터만 기준으로 캐싱하도록 구성해야 합니다. 이렇게 하면 CloudFront에서 캐시로부터 요청을 제공할 수 있는 가능성을 높이고, 그에 따라 성능이 향상되고 오리진에 걸리는 부하를 줄여 줍니다.

파라미터를 항상 동일한 순서로 나열

쿼리 문자열에서 파라미터의 순서는 중요합니다. 다음 예에서 쿼리 문자열은 파라미터 순서가 다른 것을 제외하고 동일합니다. 이 경우 CloudFront는 image.jpg에 대한 별도의 두 가지 요청을 오리진에 전달하고 객체의 두 가지 별도 버전을 캐싱합니다.

- `http://d111111abcdef8.cloudfront.net/images/image.jpg?color=red&size=large`
- `http://d111111abcdef8.cloudfront.net/images/image.jpg?size=large&color=red`

사전순과 같이 파라미터 이름을 항상 동일한 순서로 나열하는 것이 좋스빈다.

파라미터 이름 및 값에 대해 항상 동일한 대소문자 사용

CloudFront는 쿼리 문자열 파라미터를 기준으로 캐싱할 때 파라미터 이름과 값의 대소문자를 고려합니다. 다음 예에서 쿼리 문자열은 파라미터 이름과 값의 대소문자가 다른 것을 제외하고 동일합니다. 이 경우 CloudFront는 image.jpg에 대한 별도의 네 가지 요청을 오리진에 전달하고 객체의 네 가지 별도 버전을 캐싱합니다.

- `http://d111111abcdef8.cloudfront.net/images/image.jpg?color=red`
- `http://d111111abcdef8.cloudfront.net/images/image.jpg?color=Red`
- `http://d111111abcdef8.cloudfront.net/images/image.jpg?Color=red`
- `http://d111111abcdef8.cloudfront.net/images/image.jpg?Color=Red`

파라미터 이름과 값을 모두 소문자로 하는 경우와 같이 대소문자를 일관적으로 사용하는 것이 좋습니다.

서명된 URL과 충돌하는 파라미터를 사용하지 않음

서명된 URL을 사용하여 콘텐츠에 대한 액세스를 제한하는 경우(신뢰할 수 있는 서명자를 배포에 추가한 경우), CloudFront에서는 나머지 URL을 오리진에 전달하기 전에 다음 쿼리 문자열 파라미터를 제거합니다.

- `Expires`
- `Key-Pair-Id`
- `Policy`
- `Signature`

서명된 URL을 사용하면서 CloudFront에서 쿼리 문자열을 오리진에 전달하도록 구성하려는 경우 자체 쿼리 문자열 파라미터를 `Expires`, `Key-Pair-Id`, `Policy` 또는 `Signature`라는 이름으로 지정할 수 없습니다.

쿼리 문자열 파라미터 및 CloudFront 액세스 로그

웹 및 RTMP 배포에서 로깅을 활성화한 경우 CloudFront는 쿼리 문자열 파라미터를 포함해 전체 URL을 로깅합니다. 웹 배포의 경우 오리진에 쿼리 문자열을 전달하도록 CloudFront를 구성했는지 여부와 관계없이 로깅합니다. CloudFront 로깅에 대한 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

쿠키를 기반으로 콘텐츠 캐싱

웹 배포의 경우 CloudFront에서는 엣지 로케이션의 객체를 캐싱할 때 쿠키를 기본적으로 고려하지 않습니다. 오리진에서 두 개의 객체를 반환하고 이들 객체는 `Set-Cookie` 헤더의 값에만 차이가 있는 경우,

CloudFront에서는 한 가지 버전의 객체만 캐싱합니다. (RTMP 배포의 경우 쿠키를 처리하도록 CloudFront를 구성할 수 없으며 CloudFront는 엣지 캐시의 쿠키를 캐싱하지 않습니다.)

> **Important**
>
> Amazon S3 및 일부 HTTP 서버에서는 쿠키를 처리하지 않습니다. 쿠키를 처리하지 않는 오리진에 쿠키를 전달하도록 CloudFront를 구성하지 마십시오. 그렇지 않으면 캐싱 가능성에 부정적인 영향을 주며 그에 따라 성능에도 영향을 미칩니다.
> 반대로 CloudFront 오리진이 특정 쿠키에 의존하는 경우 해당 쿠키를 오리진에 전달하도록 CloudFront를 구성해야 합니다. 그렇지 않을 경우 CloudFront는 오리진에 요청을 전달하기 전에 쿠키를 제거합니다.
> 쿠키 전달을 구성하려면 배포의 캐시 동작을 업데이트합니다. 캐시 동작에 대한 자세한 내용은 캐시 동작 설정 (p. 43) 단원에서 특히 쿠키 전달 및 화이트리스트 쿠키를 참조하십시오.

최종 사용자 요청의 쿠키 일부 또는 전체를 오리진으로 전송하고, 최종 사용자 요청의 쿠키 값에 따라 별도의 객체 버전을 캐싱하도록 CloudFront를 구성할 수 있습니다. CloudFront에서는 최종 사용자 요청의 쿠키를 사용하여 캐시의 객체를 고유하게 식별합니다.

예를 들어, `locations.html`에 대한 요청에 `country` 또는 `uk` 값을 가진 `fr` 쿠키가 포함되어 있다고 가정합니다. `country` 쿠키 값을 기반으로 객체를 캐싱하도록 CloudFront를 구성할 경우, CloudFront에서는 `locations.html`에 대한 요청을 오리진에 전달하고 `country` 쿠키 및 쿠키 값을 포함합니다. 오리진에서는 `locations.html`을 반환하고 CloudFront에서는 `country` 쿠키의 값이 `uk`인 요청에 대해 한 번, 값이 `fr`인 요청에 대해 한 번, 객체를 캐싱합니다.

> **Note**
>
> 오리진에 쿠키를 전달하도록 CloudFront를 구성한 경우, CloudFront에서는 쿠키 값에 따라 캐싱합니다. 이는 오리진에서 요청의 쿠키 값을 무시하는 경우와 앞선 예제에서 항상 `locations.html`의 동일한 버전을 CloudFront에 반환하는 경우라도 마찬가지입니다. 결과적으로 CloudFront에서는 동일한 객체에 대해 더 많은 요청을 오리진 서버에 전달하게 되며, 이는 성능을 저하시키고 오리진 서버에 걸리는 부하를 가중시킵니다. 오리진 서버의 응답이 지정된 쿠키의 값에 따라 달라지지 않는 경우, 오리진에 해당 쿠키를 전달하도록 CloudFront를 구성하지 않는 것이 좋습니다.

다음 중 하나를 수행하도록 웹 배포의 각 캐시 동작을 구성할 수 있습니다.

- 오리진에 전체 쿠키 전달 – CloudFront에서 전체 쿠키를 포함한 최종 사용자 요청을 오리진에 전달합니다. 오리진에서 응답을 반환할 때 CloudFront에서는 응답과 함께 최종 사용자 요청의 쿠키 및 쿠키 값을 캐싱합니다. (오리진에서 최종 사용자 요청에 없던 쿠키를 반환하는 경우, CloudFront에서는 이를 캐싱하지 않습니다.) CloudFront에서는 최종 사용자에게 요청한 객체와 함께 최종 사용자 요청에는 없던 쿠키를 포함한 전체 쿠키 및 쿠키 값을 반환합니다.
- 지정한 쿠키의 화이트리스트 전달 – CloudFront에서는 요청을 오리진에 전달하기 전에 화이트리스트에 없는 모든 쿠키를 제거합니다. CloudFront에서는 오리진에서 보낸 응답과 함께 지정된 쿠키 및 쿠키 값을 캐싱합니다. (오리진에서 화이트리스트 지정 쿠키와 화이트리스트에 없는 쿠키를 모두 반환하는 경우 CloudFront에서는 화이트리스트 지정 쿠키만 캐싱합니다.) CloudFront에서는 최종 사용자에게 지정된 쿠키 및 쿠키 값을 포함해 객체를 반환합니다. 오리진에서 온 응답에 화이트리스트에 없는 쿠키가 포함된 경우, CloudFront에서는 그러한 쿠키 역시 최종 사용자에게 반환합니다.

  쿠키 이름에 와일드카드를 지정하는 방법에 대한 자세한 내용은 화이트리스트 쿠키 (p. 48) 단원을 참조하십시오.

  각 캐시 동작에 대해 화이트리스트로 지정할 수 있는 쿠키 이름 수에 대한 현재 할당량을 살펴보고 더 높은 할당량을 요청하려면, 화이트리스트에 등록된 쿼리 문자열에 대한 할당량(웹 배포만 해당) (p. 449) 단원을 참조하십시오.
- 오리진에 쿠키를 전달하지 않음 – CloudFront에서는 쿠키 값에 따라 객체를 캐싱하지 않습니다. 또한 CloudFront에서는 오리진에 전달하는 요청에서 `Cookie` 헤더를 제거하고 최종 사용자에게 반환하는 응답에서 `Set-Cookie` 헤더를 제거합니다.

전달하려는 쿠키를 지정할 때는 다음 사항에 유의하십시오.

액세스 로그

요청과 쿠키를 로깅하도록 CloudFront를 구성하는 경우, 쿠키를 오리진에 전달하지 않도록 CloudFront를 구성한 경우나 지정된 목록의 쿠키만 전달하도록 CloudFront를 구성한 경우라도 CloudFront에서는 전체 쿠키와 전체 쿠키 속성을 로깅합니다. CloudFront 로깅에 대한 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

대소문자 구분

쿠키 이름과 값은 모두 대소문자를 구분합니다. 예를 들어 동일한 객체에 대한 두 개의 쿠키가 대소문자를 제외하고는 동일한 경우, CloudFront에서는 이 객체를 두 번 캐싱합니다.

CloudFront에서 쿠키 정렬

CloudFront에서는 요청을 오리진에 전달하기 전에 쿠키 이름을 기준으로 쿠키를 일반 순서로 정렬합니다.

If-Modified-Since 및 If-None-Match

`If-Modified-Since` 및 `If-None-Match` 조건적 요청은 지원되지 않습니다.

표준 이름-값 페어 형식 필요

CloudFront에서는 값이 표준 이름-값 페어 형식을 따를 경우에만 쿠키 헤더를 전달합니다(예: `"Cookie: key=value; sample=cookie"`).

쿠키 값 기반 캐싱 기능 일시 중지

CloudFront에서 임시로 쿠키 및 쿠키 속성의 캐싱을 중지하려는 경우 다음 헤더를 CloudFront에 대한 응답에 추가하도록 오리진 서버를 구성합니다.

```
no-cache="Set-Cookie"
```

쿠키 이름의 총 길이

CloudFront에서 오리진으로 전달하도록 구성하는 모든 쿠키 이름의 총 바이트 수는 512에서 전달하는 쿠키 수를 뺀 값을 초과할 수 없습니다. 예를 들어 오리진에 10개의 쿠키를 전달하도록 CloudFront를 구성하는 경우, 이 10개의 쿠키 이름들을 합친 길이는 502바이트(512 – 10)를 초과할 수 없습니다. 오리진에 전체 쿠키를 전달하도록 CloudFront를 구성하는 경우, 쿠키 이름의 길이는 문제가 되지 않습니다.

CloudFront에서 오리진에 쿠키를 전달하도록 CloudFront 콘솔을 사용하여 웹 배포를 업데이트하는 방법에 대한 자세한 내용은 배포 업데이트 (p. 59) 단원을 참조하십시오. CloudFront API를 사용한 배포 업데이트에 대한 자세한 내용은 Amazon CloudFront API Reference의 UpdateDistribution을 참조하십시오.

요청 헤더에 따라 콘텐츠 캐싱

CloudFront에서 웹 배포를 할 때 CloudFront에서 헤더를 오리진으로 전송할지, 최종 사용자 요청의 헤더 값에 따라 지정된 객체의 개별 버전을 캐싱할지 여부를 선택할 수 있습니다. 이렇게 하면 사용자가 사용하는 디바이스, 최종 사용자의 위치, 최종 사용자가 사용하는 언어 및 다양한 조건에 따라 서로 다른 버전의 콘텐츠를 제공할 수 있습니다. RTMP 배포의 경우 헤더 값에 따라 캐싱하도록 CloudFront를 구성할 수 없습니다.

Note

RTMP 배포의 경우 최종 사용자 요청의 헤더에 따라 콘텐츠를 캐싱하도록 CloudFront를 구성할 수 없습니다.

주제

- 헤더 및 배포 - 개요 (p. 201)
- 캐싱의 기반이 되는 헤더 선택 (p. 201)
- CORS 설정을 존중하도록 CloudFront 구성 (p. 202)

- 디바이스 유형을 기반으로 캐싱하도록 구성 (p. 203)
- 최종 사용자 언어를 기반으로 캐싱하도록 구성 (p. 203)
- 최종 사용자 위치를 기반으로 캐싱하도록 구성 (p. 203)
- 요청의 프로토콜을 기반으로 캐싱하도록 구성 (p. 203)
- 압축 파일에 대한 캐싱 구성 (p. 203)
- 헤더를 기반으로 한 캐싱이 성능에 미치는 영향 (p. 203)
- 헤더 및 헤더 값의 대소문자가 캐싱이 미치는 영향 (p. 203)
- CloudFront에서 최종 사용자에게 반환하는 헤더 (p. 204)

헤더 및 배포 - 개요

기본적으로 CloudFront에서는 엣지 로케이션의 객체를 캐싱할 때 헤더를 고려하지 않습니다. 오리진에서 두 개의 객체를 반환하고 이들 객체는 요청 헤더의 값에만 차이가 있는 경우, CloudFront에서는 한 가지 버전의 객체만 캐싱합니다.

오리진에 헤더를 전달하도록 CloudFront를 구성할 수 있으며, 이러한 경우 CloudFront에서는 하나 이상의 요청 헤더 값에 따라 여러 버전의 객체를 캐싱합니다. 특정 헤더의 값에 따라 객체를 캐싱하기 위한 CloudFront를 구성하려면, 배포에 대한 캐시 동작 설정을 지정해야 합니다. 자세한 내용은 선택한 요청 헤더 기반의 캐시 단원을 참조하십시오.

예를 들어, `logo.jpg`에 대한 최종 사용자 요청에 `Product` 또는 `Acme` 값을 가진 사용자 지정 `Apex` 헤더가 포함되어 있다고 가정합니다. `Product` 헤더 값을 기반으로 객체를 캐싱하도록 CloudFront를 구성하는 경우 CloudFront에서는 `logo.jpg`에 대한 요청을 오리진으로 전달하고, `Product` 헤더와 헤더 값을 포함합니다. CloudFront는 `Product` 헤더의 값이 `Acme`인 요청과 값이 `Apex`인 요청에 대해 한번씩 `logo.jpg`를 캐싱합니다.

다음 중 하나를 수행하도록 웹 배포의 각 캐시 동작을 구성할 수 있습니다.

- 오리진에 전체 헤더를 전달합니다.

  > **Important**
  >
  > 전체 헤더를 오리진에 전달하도록 CloudFront를 구성한 경우, CloudFront에서는 이 캐시 동작과 연결된 객체를 캐싱하지 않습니다. 대신 각 요청을 오리진에 보냅니다.
- 지정한 헤더의 화이트리스트를 전달합니다. CloudFront에서는 모든 지정된 헤더의 값을 기반으로 객체를 캐싱합니다. 또한 CloudFront에서는 기본적으로 전달하는 헤더를 모두 전달하지만 지정한 헤더를 기반으로만 객체를 캐싱합니다.
- 기본 헤더만 전달합니다. 이 구성의 경우 CloudFront에서는 요청 헤더의 값을 기반으로 객체를 캐싱하지 않습니다.

각 캐시 동작에 대해 화이트리스트로 지정할 수 있는 헤더 수에 대한 현재 할당량을 살펴보고 더 높은 할당량을 요청하려면, 사용자 지정 헤더에 대한 할당량(웹 배포만 해당) (p. 449) 단원을 참조하십시오.

CloudFront에서 오리진에 헤더를 전달하도록 CloudFront 콘솔을 사용하여 웹 배포를 업데이트하는 방법에 대한 자세한 내용은 배포 업데이트 (p. 59) 단원을 참조하십시오. CloudFront API를 사용한 기존 배포 업데이트에 대한 자세한 내용은 Amazon CloudFront API Reference의 배포 업데이트 단원을 참조하십시오.

캐싱의 기반이 되는 헤더 선택

오리진에 전달할 수 있고 CloudFront에서 캐싱의 기반이 되는 헤더는 오리진이 Amazon S3 버킷인지 사용자 지정 오리진인지에 따라 달라집니다.

- Amazon S3 – 다수의 특정 헤더를 기반으로 객체를 전달하고 캐싱하도록 CloudFront를 구성할 수 있습니다(아래 예외 목록 참조). 하지만 cross-origin 리소스 공유(CORS)를 구현해야 하거나 오리진 방향 이벤트

에서 Lambda@Edge를 사용하여 콘텐츠를 개인 설정하고 싶은 경우가 아니라면 Amazon S3 오리진을 통해 헤더를 화이트리스트에 등록하지 않는 것이 좋습니다.

- CORS를 구성하려면 CloudFront가 CORS에 대해 활성화된 웹 사이트용 콘텐츠를 배포할 수 있도록 헤더를 전달해야 합니다. 자세한 내용은 CORS 설정을 존중하도록 CloudFront 구성 (p. 202) 단원을 참조하십시오.
- Amazon S3 오리진에 전달된 헤더를 사용하여 콘텐츠를 개인 설정하려면 Lambda@Edge 함수를 작성 및 추가하고 이를 오리진 방향 이벤트에서 트리거되는 CloudFront 배포에 연결합니다. 콘텐츠를 개인 설정하기 위해 헤더를 처리하는 방법에 대한 자세한 내용은 국가 또는 디바이스 유형 헤더에 따른 콘텐츠 개인 설정 - 예제 (p. 341) 단원을 참조하십시오.

  콘텐츠를 개인 설정하는 데 사용하지 않는 헤더들은 화이트리스트에 등록하지 않는 것이 좋습니다. 왜냐하면 추가 헤더를 전달하면 캐시 적중률이 낮아질 수 있기 때문입니다. 즉, CloudFront는 모든 요청의 비율로서 엣지 캐시에서 제공되는 다수의 요청을 처리할 수 없습니다.

- 사용자 지정 오리진 – 다음과 같이 요청 헤더 값을 기반으로 캐싱하도록 CloudFront를 구성할 수 있습니다.
  - `Connection`
  - `Cookie` – 쿠키를 기반으로 전달 및 캐싱을 수행하려는 경우, 배포에 개별 설정을 사용합니다. 자세한 내용은 쿠키를 기반으로 콘텐츠 캐싱 (p. 198) 단원을 참조하십시오.
  - `Host (for Amazon S3 origins)`
  - `Proxy-Authorization`
  - `TE`
  - `Upgrade`

  `Date` 및 `User-Agent` 헤더의 값에 따라 객체를 캐싱하도록 CloudFront를 구성할 수 있지만, 권장되지는 않습니다. 이러한 헤더는 여러 가지 값을 가질 수 있으며, 헤더의 값에 따른 캐싱으로 인해 CloudFront에서 오리진에 전달되는 요청이 눈에 띄게 증가할 수 있습니다.

HTTP 요청 헤더의 전체 목록과 CloudFront에서 이를 처리하는 방법은 HTTP 요청 헤더 및 CloudFront 동작 (사용자 지정 및 S3 오리진) (p. 239) 단원을 참조하십시오.

CORS 설정을 존중하도록 CloudFront 구성

Amazon S3 버킷 또는 사용자 지정 오리진에서 CORS를 활성화한 경우에는 CORS 설정을 준수하도록 전달할 특정 헤더를 선택해야 합니다. 오리진(Amazon S3 또는 사용자 지정)과 `OPTIONS` 응답의 캐싱 여부에 따라 서로 다른 헤더를 전달해야 합니다.

Amazon S3

- `OPTIONS` 응답을 캐싱하고 싶으면 다음을 수행하세요.
  - `OPTIONS` 응답에 대해 캐싱을 활성화하는 기본 캐시 동작 설정에 대한 옵션을 선택합니다.
  - `Origin`, `Access-Control-Request-Headers` 및 `Access-Control-Request-Method` 헤더를 전달하도록 CloudFront를 구성합니다.
- `OPTIONS` 응답을 캐싱하고 싶지 않으면 오리진에서 요청된 다른 모든 헤더와 함께 `Origin` 헤더를 전달하도록 CloudFront를 구성합니다(예: `Access-Control-Request-Headers`, `Access-Control-Request-Method`, 또는 기타).

사용자 지정 오리진 – 오리진에서 요구하는 나머지 헤더와 함께 `Origin` 헤더를 전달합니다.

CloudFront 배포를 위해 캐시 동작에서 헤더를 화이트리스트로 만들어 헤더를 전달하도록 CloudFront를 구성합니다. 헤더 전달 작업에 대한 자세한 내용은 헤더 및 배포 - 개요 (p. 201) 단원을 참조하십시오.

CORS 및 Amazon S3에 대한 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 Cross-Origin 리소스 공유 활성화 단원을 참조하십시오.

디바이스 유형을 기반으로 캐싱하도록 구성

CloudFront에서 사용자가 콘텐츠를 보기 위해 사용 중인 디바이스에 따라 여러 버전의 객체를 캐싱하려는 경우, 다음과 같이 적용 가능한 헤더를 사용자 지정 오리진에 전달하도록 CloudFront를 구성합니다.

- `CloudFront-Is-Desktop-Viewer`
- `CloudFront-Is-Mobile-Viewer`
- `CloudFront-Is-SmartTV-Viewer`
- `CloudFront-Is-Tablet-Viewer`

`User-Agent` 헤더의 값에 따라 CloudFront에서는 요청을 오리진에 전달하기 전에 이러한 헤더의 값을 `true` 또는 `false`로 설정합니다. 디바이스가 둘 이상의 범주에 해당하는 경우 둘 이상의 값이 `true`일 수 있습니다. 예를 들어, 일부 태블릿 디바이스의 경우 CloudFront에서는 `CloudFront-Is-Mobile-Viewer`와 `CloudFront-Is-Tablet-Viewer`를 모두 `true`로 설정할 수 있습니다.

최종 사용자 언어를 기반으로 캐싱하도록 구성

CloudFront에서 요청에 지정된 언어에 따라 여러 버전의 객체를 캐싱하려는 경우, `Accept-Language` 헤더의 언어를 포함하도록 애플리케이션을 프로그래밍하고 `Accept-Language` 헤더를 오리진에 전달하도록 CloudFront를 구성합니다.

최종 사용자 위치를 기반으로 캐싱하도록 구성

CloudFront에서 요청이 시작된 국가에 따라 여러 버전의 객체를 캐싱하려는 경우, `CloudFront-Viewer-Country` 헤더를 오리진에 전달하도록CloudFront를 구성합니다. CloudFront에서는 자동으로 요청이 시작된 IP 주소를 두 글자의 국가 코드로 변환합니다. 코드와 국가 이름으로 정렬이 가능하고 사용이 간편한 국가 코드 목록에 대해서는 Wikipedia 항목 ISO 3166-1 alpha-2를 참조하십시오.

요청의 프로토콜을 기반으로 캐싱하도록 구성

CloudFront에서 HTTP 또는 HTTPS 중 요청의 프로토콜에 따라 여러 버전의 객체를 캐싱하려는 경우, `CloudFront-Forwarded-Proto` 헤더를 오리진에 전달하도록 CloudFront를 구성합니다.

압축 파일에 대한 캐싱 구성

오리진이 Brotli 압축을 지원할 경우, `Accept-Encoding` 헤더를 화이트리스트로 지정하여 이 헤더를 기반으로 캐시할 수 있습니다. 오리진이 이 헤더를 기반으로 하여 다른 콘텐츠를 제공하는 경우에만 `Accept-Encoding`을 기반으로 캐싱하도록 구성해야 합니다.

헤더를 기반으로 한 캐싱이 성능에 미치는 영향

하나 이상의 헤더를 기반으로 캐싱하도록 CloudFront를 구성하고 그러한 헤더가 둘 이상의 값을 가질 수 있는 경우, CloudFront에서는 동일한 객체에 대해 더 많은 요청을 오리진 서버에 전달합니다. 이는 성능을 저하시키고 오리진 서버에 걸리는 부하를 가중시킵니다. 오리진 서버에서 지정된 헤더의 값에 상관없이 동일한 객체를 반환하는 경우, 해당 헤더를 기반으로 캐싱하도록 CloudFront를 구성하지 않는 것이 좋습니다.

둘 이상의 헤더를 전달하도록 CloudFront를 구성하는 경우, 최종 사용자 요청의 헤더 순서는 헤더의 값이 동일하다면 캐싱에 영향을 주지 않습니다. 예를 들어 하나의 요청에 A:1,B:2 헤더가 포함되어 있고 다른 요청에는 B:2,A:1이 포함되어 있는 경우, CloudFront에서는 하나의 객체 사본만 캐싱합니다.

헤더 및 헤더 값의 대소문자가 캐싱이 미치는 영향

CloudFront에서 헤더 값에 따라 캐싱을 수행할 경우 헤더 이름의 대소문자는 구별하지 않지만 헤더 값의 대소문자는 구별합니다.

- 최종 사용자 요청에 `Product:Acme` 및 `product:Acme`가 모두 포함된 경우, CloudFront에서는 객체를 한 번만 캐싱합니다. 이들은 헤더 이름의 대소문자만 다른 경우로, 캐싱에는 영향을 미치지 않습니다.
- 최종 사용자 요청에 `Product:Acme` 및 `Product:acme`가 모두 포함된 경우 CloudFront에서는 객체를 두 번 캐싱합니다. 이는 일부 요청에서는 값이 `Acme`이고 다른 요청에서는 `acme`이기 때문입니다.

CloudFront에서 최종 사용자에게 반환하는 헤더

특정 헤더를 전달하고 캐싱하도록 CloudFront를 구성하더라도 CloudFront에서 최종 사용자에게 반환하는 헤더의 종류에 영향을 미치지 않습니다. CloudFront에서는 오리진에서 가져온 헤더를 모두 반환하며, 여기에는 몇 가지 예외 사항이 적용됩니다. 자세한 내용은 관련 주제를 참조하십시오.

- Amazon S3 오리진 – CloudFront에서 제거하거나 업데이트하는 HTTP 응답 헤더 (p. 235) 단원을 참조하십시오.
- 사용자 지정 오리진 – CloudFront에서 제거하거나 대체하는 HTTP 응답 헤더 (p. 246) 단원을 참조하십시오.

콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료)

CloudFront에서 다른 요청을 오리진에 전달하기 전에 파일을 CloudFront 캐시에 보관하는 시간을 제어할 수 있습니다. 이 기간을 단축함으로써 동적 콘텐츠를 제공할 수 있습니다. 이 기간이 늘어나면 파일이 엣지 캐시에서 바로 제공될 가능성이 높으므로 사용자에게 제공되는 성능이 향상됩니다. 보관 기간이 늘어나면 오리진에 걸리는 부하 역시 줄어듭니다.

일반적으로 CloudFront는 지정한 캐시 기간이 지날 때까지—즉 파일이 만료될 때까지 엣지 로케이션에서 파일을 제공합니다. 파일이 만료된 후 다음에 엣지 로케이션에서 이 파일에 대한 사용자 요청을 받으면, CloudFront에서는 이 요청을 오리진 서버로 전달하여 캐시에 최신 버전의 파일이 포함되어 있는지 확인합니다. 오리진의 응답은 파일 변경 여부에 따라 달라집니다.

- CloudFront 캐시에 이미 최신 버전이 있는 경우 오리진에서는 304 상태 코드(수정되지 않음)를 반환합니다.
- CloudFront 캐시에 최신 버전이 없는 경우 오리진에서는 200 상태 코드(OK)와 최신 버전의 파일을 반환합니다.

엣지 로케이션의 파일이 자주 요청되지 않는 경우—CloudFront에서는 만료 날짜 전에 이 파일을 제거하여—보다 최근에 요청된 파일을 위한 공간을 마련할 수 있습니다.

기본적으로 각 파일은 24시간 후에 자동으로 만료되지만, 다음 두 가지 방식으로 기본 동작을 변경할 수 있습니다.

- 동일한 경로 패턴과 일치하는 모든 파일에 대해 캐시 기간을 변경하려면 캐시 동작에 대한 Minimum TTL(최소 TTL), Maximum TTL(최대 TTL) 및 Default TTL(기본 TTL)의 CloudFront 설정을 변경할 수 있습니다. 개별 설정에 대한 자세한 내용은 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35)의 최소 TTL, 최대 TTL 및 기본 TTL을 참조하십시오. 이러한 설정을 사용하려면 CloudFront 배포를 생성하거나 업데이트할 때 Object Caching(객체 캐싱) 설정의 사용자 지정 옵션을 선택해야 합니다. 자세한 내용은 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35)에서 객체 캐싱을 참조하십시오.
- 개별 파일의 캐시 기간을 변경하려면 `Cache-Control max-age` 또는 `Cache-Control s-maxage` 명령 또는 `Expires` 헤더 필드를 파일에 추가하도록 오리진을 구성할 수 있습니다. 자세한 내용은 헤더를 사용하여 개별 객체의 캐시 기간 제어 (p. 205) 단원을 참조하십시오.

최소 TTL, 기본 TTL 및 최대 TTL이 `Cache-Control max-age` 및 `Cache-Control s-maxage` 명령 및 `Expires` 헤더 필드와 상호 작용하는 방식에 대한 자세한 내용은 CloudFront에서 웹 배포용 객체를 캐싱하는 시간 지정 (p. 206) 단원을 참조하십시오.

또한 CloudFront에서 다른 요청을 오리진에 전달하여 요청된 객체의 가져오기를 다시 시도하기 전에 오류(예: 404, 찾을 수 없음)가 CloudFront 캐시에 보관되는 시간 역시 제어할 수 있습니다. 자세한 내용은 CloudFront에서 오리진의 HTTP 4xx 및 5xx 상태 코드를 처리 및 캐싱하는 방법 (p. 250) 단원을 참조하십시오.

주제

- 헤더를 사용하여 개별 객체의 캐시 기간 제어 (p. 205)
- CloudFront에서 웹 배포용 객체를 캐싱하는 시간 지정 (p. 206)
- CloudFront에서 RTMP 배포용 객체를 캐싱하는 최소 시간 지정 (p. 208)
- Amazon S3 콘솔을 사용하여 객체에 헤더 추가 (p. 209)

헤더를 사용하여 개별 객체의 캐시 기간 제어

`Cache-Control` 및 `Expires` 헤더를 사용하여 객체가 캐시에 머무르는 시간을 제어할 수 있습니다. 최소 TTL, 기본 TTL 및 최대 TTL에 대한 설정은 캐시 기간에도 영향을 주지만 캐시 기간에 영향을 미치는 방식에 대한 개요는 다음과 같습니다.

- `Cache-Control max-age` 명령을 사용하면 CloudFront에서 오리진 서버로부터 객체를 가져오기 전에 이 객체를 캐시에 보관하려는 시간(초)을 지정할 수 있습니다. CloudFront에서 지원되는 최소 만료 시간은 웹 배포의 경우 0초, RTMP 배포의 경우 3,600초입니다. 최대값은 100년입니다. 다음 형식으로 값을 지정합니다.

  ```
  Cache-Control: max-age=#
  ```

  예를 들어, 다음 명령은 CloudFront에 3,600초(1시간) 동안 관련 객체를 캐시에 보관하도록 지시합니다.

  ```
  Cache-Control: max-age=3600
  ```

  브라우저 캐시에 보관하는 기간과는 다른 기간 동안 객체를 CloudFront 엣지 캐시에 보관하려는 경우, `Cache-Control max-age` 및 `Cache-Control s-maxage` 명령을 함께 사용할 수 있습니다. 자세한 내용은 CloudFront에서 웹 배포용 객체를 캐싱하는 시간 지정 (p. 206) 단원을 참조하십시오.
- `Expires` 헤더 필드를 사용하면 다음과 같이 RFC 2616, Hypertext Transfer Protocol -- HTTP/1.1 Section 3.3.1, Full Date에 지정된 형식을 사용하여 만료 날짜와 시간을 지정할 수 있습니다.

  ```
  Sat, 27 Jun 2015 23:59:59 GMT
  ```

`Expires` 헤더 필드 대신 `Cache-Control max-age` 명령을 사용하여 객체 캐싱을 제어하는 것이 좋습니다. `Cache-Control max-age` 및 `Expires` 모두에 대해 값을 지정한 경우, CloudFront에서는 `Cache-Control max-age`의 값만 사용합니다.

자세한 내용은 CloudFront에서 웹 배포용 객체를 캐싱하는 시간 지정 (p. 206) 단원을 참조하십시오.

최종 사용자의 `GET` 요청에 HTTP `Cache-Control` 또는 `Pragma` 헤더 필드를 사용하여 CloudFront가 객체에 대한 오리진 서버로 돌아가도록 할 수는 없습니다. CloudFront에서는 최종 사용자 요청의 헤더 필드를 무시합니다.

`Cache-Control` 및 `Expires` 헤더 필드에 대한 자세한 내용은 RFC 2616, Hypertext Transfer Protocol -- HTTP/1.1의 다음 단원을 참조하십시오.

- 섹션 14.9 Cache Control
- 섹션 14.21 Expires

PHP용 AWS SDK를 사용하여 `Cache-Control` 및 `Expires` 헤더 필드를 추가하는 방법의 예시는 Amazon Simple Storage Service 개발자 가이드의 Upload an Object Using the AWS SDK for PHP 단원을 참조하십시오. 일부 타사 도구를 사용하여 이러한 필드를 추가할 수도 있습니다.

CloudFront에서 웹 배포용 객체를 캐싱하는 시간 지정

웹 배포에서 `Cache-Control` 또는 `Expires` 헤더 및 CloudFront 최소, 최대 및 기본 TTL 값을 사용하여 다른 요청을 오리진에 전달하기 전에 CloudFront가 캐시에 객체를 보관하는 시간(초)를 제어할 수 있습니다. 헤더 값은 다른 요청을 CloudFront에 전달하기 전에 브라우저에서 객체를 캐시에 보관하는 시간도 결정합니다.

> **Important**
>
> 하나의 캐시 동작에 대해 전체 헤더를 오리진에 전달하도록 CloudFront를 구성한 경우, CloudFront에서는 연결된 객체를 캐시하지 않습니다. 대신 CloudFront에서는 그러한 객체에 대한 전체 요청을 오리진에 전달합니다. 그 구성에서 최소 TTL 값은 0이어야 합니다. 자세한 내용은 요청 헤더에 따라 콘텐츠 캐싱 (p. 200) 단원을 참조하십시오.

Minimum TTL(최소 TTL), Maximum TTL(최대 TTL) 및 Default TTL(기본 TTL)의 값을 지정하려면 Object Caching(객체 캐싱) 설정에 대해 사용자 지정 옵션을 선택해야 합니다.

오리진 구성	최소 TTL = 0초	최소 TTL > 0초
오리진에서 **`Cache-Control max-age`** 명령을 객체에 추가	CloudFront 캐싱 CloudFront는 `Cache-Control max-age` 명령의 값 또는 CloudFront 최대 TTL의 값 중 더 작은 값 동안 객체를 캐싱합니다. 브라우저 캐싱 브라우저는 `Cache-Control max-age` 명령의 값 동안 객체를 캐싱합니다.	CloudFront 캐싱 CloudFront 캐싱은 CloudFront 최소 TTL과 최대 TTL 및 `Cache-Control max-age` 명령의 값에 따라 달라집니다. • 최소 TTL < `max-age` < 최대 TTL CloudFront는 `Cache-Control max-age` 명령의 값 동안 객체를 캐싱합니다. • `max-age` < 최소 TTL CloudFront는 CloudFront 최소 TTL의 값 동안 객체를 캐싱합니다. • `max-age` > 최대 TTL CloudFront는 CloudFront 최대 TTL의 값 동안 객체를 캐싱합니다. 브라우저 캐싱 브라우저는 `Cache-Control max-age` 명령의 값 동안 객체를 캐싱합니다.
오리진에서 **`Cache-Control max-age`** 명령을 객체에 추가하지 않음	CloudFront 캐싱 CloudFront는 CloudFront 기본 TTL의 값 동안 객체를 캐싱합니다. 브라우저 캐싱 브라우저에 따라 다릅니다.	CloudFront 캐싱 CloudFront는 CloudFront 최소 TTL 또는 기본 TTL의 값 중 더 큰 값 동안 객체를 캐싱합니다. 브라우저 캐싱 브라우저에 따라 다릅니다.

오리진 구성	최소 TTL = 0초	최소 TTL > 0초
오리진에서 **`Cache-Control max-age`** 및 **`Cache-Control s-maxage`** 명령을 객체에 추가	CloudFront 캐싱 CloudFront는 `Cache-Control s-maxage` 명령의 값 또는 CloudFront 최대 TTL의 값 중 더 작은 값 동안 객체를 캐싱합니다. 브라우저 캐싱 브라우저는 `Cache-Control max-age` 명령의 값 동안 객체를 캐싱합니다.	CloudFront 캐싱 CloudFront 캐싱은 CloudFront 최소 TTL과 최대 TTL 및 `Cache-Control s-maxage` 명령의 값에 따라 달라집니다. • 최소 TTL < `s-maxage` < 최대 TTL CloudFront는 `Cache-Control s-maxage` 명령의 값 동안 객체를 캐싱합니다. • `s-maxage` < 최소 TTL CloudFront는 CloudFront 최소 TTL의 값 동안 객체를 캐싱합니다. • `s-maxage` > 최대 TTL CloudFront는 CloudFront 최대 TTL의 값 동안 객체를 캐싱합니다. 브라우저 캐싱 브라우저는 `Cache-Control max-age` 명령의 값 동안 객체를 캐싱합니다.

오리진 구성	최소 TTL = 0초	최소 TTL > 0초
오리진에서 **`Expires`** 헤더를 객체에 추가	CloudFront 캐싱 CloudFront는 `Expires` 헤더의 날짜 또는 CloudFront 최대 TTL의 값 중 더 빨리 도래하는 때까지 객체를 캐싱합니다. 브라우저 캐싱 브라우저는 `Expires` 헤더의 날짜까지 객체를 캐싱합니다.	CloudFront 캐싱 CloudFront 캐싱은 CloudFront 최소 TTL과 최대 TTL 및 `Expires` 헤더의 값에 따라 달라집니다. • 최소 TTL < `Expires` < 최대 TTL CloudFront는 `Expires` 헤더의 날짜와 시간까지 객체를 캐싱합니다. • `Expires` < 최소 TTL CloudFront는 CloudFront 최소 TTL의 값 동안 객체를 캐싱합니다. • `Expires` > 최대 TTL CloudFront는 CloudFront 최대 TTL의 값 동안 객체를 캐싱합니다. 브라우저 캐싱 브라우저는 `Expires` 헤더의 날짜와 시간까지 객체를 캐싱합니다.
오리진에서 **`Cache-Control: no-cache`**, **`no-store`** 및/또는 **`private`** 명령을 객체에 추가	CloudFront 및 브라우저에서는 헤더의 설정을 따릅니다. CloudFront에서 `Cache-Control: no-cache` 헤더를 처리하는 방식에 대한 예외 사항은 동일 객체에 대한 동시 요청(트래픽 스파이크) (p. 244) 단원을 참조하십시오.	CloudFront 캐싱 CloudFront는 CloudFront 최소 TTL의 값 동안 객체를 캐싱합니다. 브라우저 캐싱 브라우저에서는 헤더의 설정을 따릅니다.

CloudFront 콘솔을 사용하여 웹 배포에 대한 설정을 변경하는 방법에 대한 자세한 내용은 배포 업데이트 (p. 59) 단원을 참조하십시오. CloudFront API를 사용하여 웹 배포에 대한 설정을 변경하는 방법에 대한 자세한 내용은 PUT Config 단원을 참조하십시오.

CloudFront에서 RTMP 배포용 객체를 캐싱하는 최소 시간 지정

RTMP 배포의 경우 CloudFront에서는 기본적으로 24시간 동안 엣지 캐시에 객체를 보관합니다. `Cache-Control` 또는 `Expires` 헤더를 객체에 추가하여 다른 요청을 오리진에 전달하기 전에 CloudFront에서 엣지 캐시에 객체를 보관하는 최소 시간을 변경할 수 있습니다. 최소 기간은 3,600초(1시간)입니다. 더 낮은 값을 지정하는 경우 CloudFront에서는 3,600초를 사용합니다.

Amazon S3 콘솔을 사용하여 객체에 헤더 추가

> **Note**
>
> Amazon S3 콘솔에서는 한 번에 하나의 객체에만 헤더를 추가할 수 있지만, 일부 타사 도구를 사용하면 한 번에 여러 Amazon S3 객체에 헤더를 추가할 수 있습니다. Amazon S3를 지원하는 타사 도구에 대한 자세한 내용은 **`AWS S3 third party tools`**를 주제로 웹 검색을 이용해 보십시오.

Amazon S3 콘솔을 사용하여 **`Cache-Control`** 또는 **`Expires`** 헤더 필드를 Amazon S3 객체에 추가하려면

1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/s3에서 Amazon S3 콘솔을 엽니다.
2. Amazon S3 콘솔의 버킷 목록에서 해당 파일이 포함된 버킷 이름을 선택합니다.
3. 이름 목록에서 헤더를 추가하려는 객체의 이름을 선택합니다.
4. 속성를 선택한 다음, 메타데이터를 선택합니다.
5. 메타데이터 추가를 선택한 다음 키 메뉴에서 Cache-Control 또는 Expires를 선택합니다.
6. 값필드에 다음 사항 중 하나를 입력합니다.
   - [`Cache-Control`] 필드에 다음을 입력합니다.

     `max-age=`*`number of seconds that you want objects to stay in a CloudFront edge cache`*
   - 만료 필드에 날짜와 시간을 HTML 형식으로 입력합니다.
7. Save를 선택합니다.

CloudFront 오리진 장애 조치를 통한 고가용성 최적화

고가용성을 필요로 하는 시나리오에 대한 오리진 장애 조치를 사용하여 이제 CloudFront를 설정할 수 있습니다. 시작 단계로써, CloudFront에 대한 기본 오리진을 지정할 오리진 그룹을 만들고, 기본 오리진이 특정 HTTP 상태 코드 실패 응답을 반환할 때 CloudFront가 자동으로 전환하는 두 번째 오리진을 지정합니다.

오리진 장애 조치를 설정하려면 최소 2개의 오리진이 있는 배포가 전제되어야 합니다. 다음으로, 2개의 오리진이 포함된 배포에 대한 오리진 그룹을 만들고 그 중 1개의 오리진을 기본 오리진으로 설정합니다. 마지막으로, 오리진 그룹을 오리진으로 지정하는 캐시 동작을 정의합니다. 배포에 대한 오리진 그룹 지정에 대한 자세한 내용은 Origin ID (p. 39) 단원을 참조하십시오.

오리진 그룹 내의 2개의 오리진은 Amazon S3 버킷이나 Amazon EC2 인스턴스와 같은 AWS 오리진 또는 자체의 HTTP 웹 서버와 같은 사용자 지정 오리진으로 이루어진 어떠한 조합이든 가능합니다. 오리진 그룹을 만들 때에는, 기본 오리진이 사용자가 구성한 특정 상태 코드를 반환할 때 CloudFront가 GET, HEAD 및 OPTIONS HTTP 메서드로 장애 조치를 취하도록 구성합니다.

특정 오리진 장애 조치 옵션과 함께 오리진 그룹을 설정하는 단계는 오리진 그룹 만들기 (p. 212) 단원을 참조하십시오.

캐시 동작에 대한 오리진 장애 조치를 구성한 후에, CloudFront가 최종 사용자의 요청에 대해 다음을 수행합니다.

- 캐시 적중인 경우, CloudFront가 요청된 파일을 반환합니다.
- 캐시 누락이 있는 경우, CloudFront가 오리진 그룹 내에서 식별해낸 기본 오리진으로 요청을 라우팅합니다.

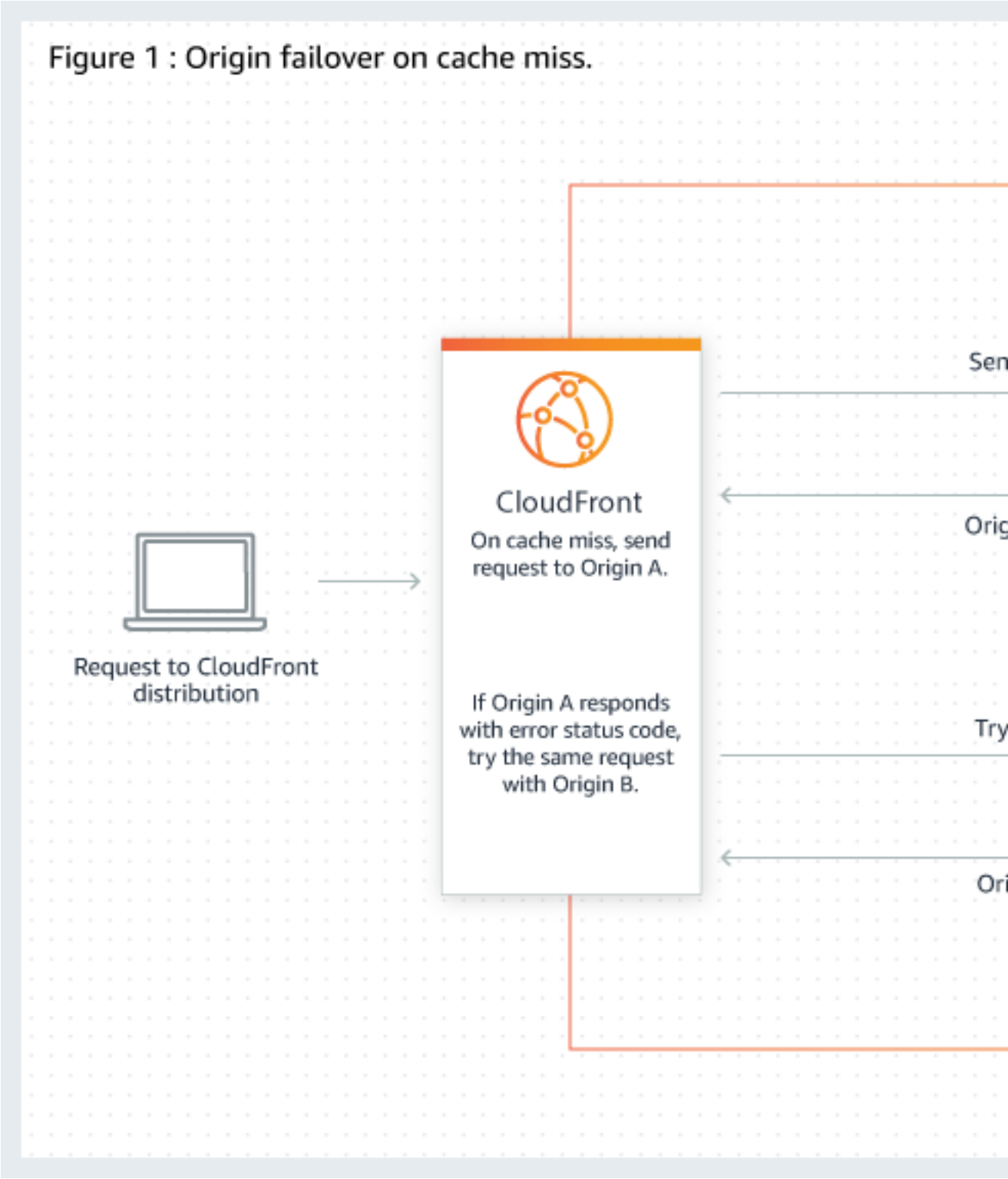
- HTTP 2xx 또는 HTTP 3xx 상태 코드 등 장애 조치에 대해 구성되지 않은 상태 코드가 반환된 경우, CloudFront가 요청된 콘텐츠를 제공합니다.
- 기본 오리진이 장애 조치에 대해 구성된 HTTP 상태 코드를 반환하거나 제한 시간이 경과한 경우, CloudFront가 해당 요청을 오리진 그룹 내의 백업 오리진으로 라우팅합니다.

  Note

  1건의 요청에 대해 3회 연속 오리진에 대한 연결을 시도했음에도 연결할 수 없는 경우, CloudFront에서 '연결 제한 시간 초과'를 발령합니다. CloudFront는 각 연결 시도 사이에 최대 10초 간 대기합니다.

앞선 요청이 두 번째 오리진으로 장애 조치된 경우에도, CloudFront는 수신되는 모든 요청을 기본 오리진으로 라우팅합니다. 기본 오리진으로의 라우팅 시도에 대하여 장애 조치 또는 제한 시간 초과에 대해 구성된 상태 코드가 반환된 후에야 CloudFront는 두 번째 오리진으로 요청을 전송합니다.

다음 다이어그램에서는 오리진 장애 조치의 작동 방식을 보여 줍니다.



자세한 정보는 다음 단원을 참조하십시오.

- 캐시 적중 수 및 누락 수: 캐싱과 CloudFront 엣지 캐시의 작동 방식 (p. 193)
- 오리진 장애 조치를 갖는 요청 및 응답 동작: 오리진 그룹에 대한 요청 및 응답 동작 (p. 247)

오리진 그룹 만들기

2개의 오리진으로 오리진 그룹을 생성합니다. 1개의 오리진을 기본 오리진으로 지정하고, 기본 오리진이 특정 HTTP 상태 코드 실패 응답을 반환할 때 CloudFront가 자동으로 전환하는 두 번째 오리진을 지정합니다.

오리진 그룹을 생성하려면

1. Origin and Origin Groups(오리진 및 오리진 그룹) 탭에서 Create origin group을 선택합니다.
2. 오리진 그룹에 대해 오리진을 선택한 후 우선 순위 화살표를 선택하여 오리진 1개를 기본 오리진으로 설정합니다.
3. 1개의 HTTP 상태 코드를 장애 조치 기준으로 선택합니다. 500, 502, 503, 504, 404 또는 403 상태 코드의 어떠한 조합도 선택할 수 있습니다.
4. 오리진 그룹에 대한 고유 설명 ID를 입력합니다. 다른 오리진이나 오리진 그룹에 이미 사용되고 있는 ID는 사용할 수 없습니다.

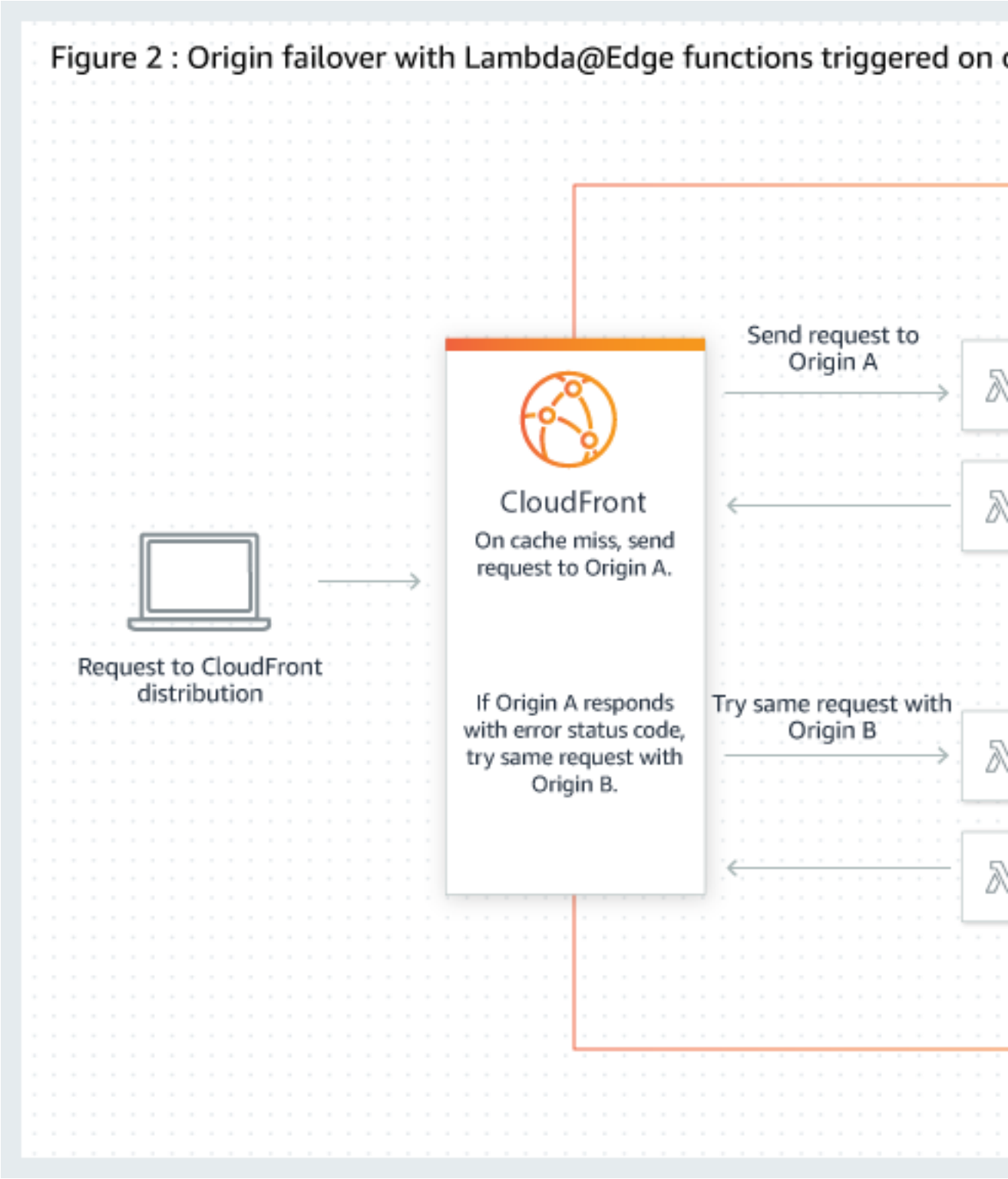
배포에 대한 오리진 그룹 지정에 대한 자세한 내용은 Origin ID (p. 39) 단원을 참조하십시오.

Lambda@Edge 함수와 함께 오리진 장애 조치 사용

Lambda@Edge 함수를 오리진 그룹에 설정한 CloudFront 배포와 함께 사용할 수 없습니다. Lambda 함수를 사용하려면, 캐시 동작을 생성할 때 오리진 그룹에 대한 오리진 요청 또는 오리진 응답 트리거에 함수를 지정합니다.

Lambda 함수는, CloudFront가 캐시 누락 시 요청을 라우팅하는 동안에는, 기본 오리진에 대해 트리거됩니다. 기본 오리진이 장애 조치에 대해 구성하지 않은 HTTP 상태 코드를 반환할 때 CloudFront가 요청을 두 번째 오리진으로 다시 라우팅하는 경우, Lambda 함수가 다시 트리거됩니다.

다음 다이어그램에서는 오리진 요청이나 응답 트리거에 Lambda@Edge 함수가 포함된 경우 오리진 장애 조치가 작동하는 방식을 보여 줍니다.



Lambda@Edge 트리거 사용에 대한 자세한 내용은 Lambda@Edge 함수에 대한 트리거 추가 (p. 302) 단원을 참조하십시오.

사용자 지정 오류 페이지와 함께 오리진 장애 조치 사용

사용자 지정 오류 페이지를 오리진 장애 조치에 대해 설정되지 않은 오리진에 사용하는 경우와 유사한 방식으로 사용자 지정 오류 페이지를 오리진 그룹에 사용할 수 있습니다.

오리진 장애 조치를 사용할 때, CloudFront를 기본 오리진 또는 두 번째 오리진에 대해 (또는 양쪽 모두에 대해) 사용자 지정 오류 페이지를 반환하도록 구성할 수 있습니다.

- 기본 오리진에 대한 사용자 지정 오류 페이지를 반환합니다. 기본 오리진에서 장애 조치에 대해 구성되지 않은 HTTP 상태 코드 오류를 반환할 경우, CloudFront는 사용자 지정 오류 페이지를 최종 사용자에게 반환합니다.
- 두 번째 오리진에 대한 사용자 지정 오류 페이지를 반환합니다. CloudFront가 두 번째 오리진으로 장애 조치를 취하는 경우, CloudFront는 사용자 지정 오류 페이지를 반환합니다.

CloudFront에서 사용자 지정 오류 페이지를 사용하는 방법에 대한 자세한 내용은 사용자 지정 오류 응답 생성 중 (p. 255) 단원을 참조하십시오.

CloudFront에서 객체에 대한 부분적인 요청을 처리하는 방법(Range GET)

대용량 객체의 경우 최종 사용자의 브라우저 또는 클라이언트에서 여러 `GET` 요청을 만들고 `Range` 요청 헤더를 사용하여 더 작은 단위로 객체를 다운로드할 수 있습니다. `Range GET` 요청이라고도 하는 이러한 바이트 범위의 요청을 통해 부분 다운로드의 효율을 높이고 부분적으로 실패한 전송을 쉽게 복구할 수 있습니다.

CloudFront에서 `Range GET` 요청을 받은 경우, 요청을 받은 엣지 로케이션에서 캐시를 확인합니다. 엣지 로케이션의 캐시에 이미 전체 객체 또는 객체의 요청된 부분이 포함되어 있는 경우, CloudFront에서는 캐시로부터 요청된 범위를 즉시 제공합니다.

캐시에 요청된 범위가 포함되어 있지 않은 경우, CloudFront에서는 요청을 오리진에 전달합니다. (성능 최적화를 위해 CloudFront에서는 클라이언트가 `Range GET`에서 요청한 것보다 더 큰 범위를 요청할 수 있습니다.) 오리진에서 `Range GET` 요청을 지원하는지 여부에 따라 다음 상황이 달라집니다.

- 오리진이 **`Range GET`** 요청을 지원하는 경우: 요청된 범위를 반환합니다. CloudFront에서는 요청된 범위를 제공하고 향후 요청을 위해 해당 범위를 캐싱합니다. (Apache 및 IIS 등과 같은 일부 HTTP 서버에서 하는 것처럼 Amazon S3에서는 `Range GET` 요청을 지원합니다. 특정 HTTP 서버에서의 지원 여부를 알아보려면 해당 HTTP 서버의 설명서를 참조하십시오.)
- 오리진이 **`Range GET`** 요청을 지원하지 않는 경우: 전체 객체를 반환합니다. CloudFront에서는 전체 객체를 제공하고 향후 요청을 위해 해당 전체 객체를 캐싱합니다. CloudFront에서 엣지 캐시에 전체 객체를 캐싱한 후에는 요청된 범위를 제공하여 `Range GET` 요청에 응답합니다.

두 경우 모두 오리진에서 첫 번째 바이트가 도착하면 CloudFront에서는 요청된 범위 또는 객체를 최종 사용자에게 제공하기 시작합니다.

> **Note**
>
> 최종 사용자가 `Range GET` 요청을 하고 오리진이 `Transfer-Encoding: chunked`를 반환하는 경우, CloudFront에서는 요청한 범위 대신에 뷰어에 전체 객체를 반환합니다.

CloudFront에서는 일반적으로 `Range` 헤더에 대한 RFC 사양을 준수합니다. 그러나 `Range` 헤더가 다음 요구사항을 준수하지 않는 경우, CloudFront에서는 지정된 범위와 함께 상태 코드 206을 반환하는 대신 전체 객체와 함께 HTTP 상태 코드 200을 반환합니다.

- 범위는 오름차순으로 나열되어야 합니다. 예를 들어, `100-200,300-400`은 유효하지만 `300-400,100-200`은 유효하지 않습니다.

- 범위는 중복되어서는 안 됩니다. 예를 들어, `100-200,150-250`은 유효하지 않습니다.
- 모든 범위 사양은 유효해야 합니다. 예를 들어, 범위의 일부로 음수 값을 지정할 수 없습니다.

`Range`요청 헤더에 대한 자세한 내용은 http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.35에서 Hypertext Transfer Protocol -- HTTP/1.1의 "섹션 14.35 Range"를 참조하십시오.

기본 루트 객체 지정

사용자가 배포의 객체가 아닌 웹 배포의 루트 URL을 요청할 경우, 특정 객체(기본 루트 객체)를 반환하도록 CloudFront를 구성할 수 있습니다. 기본 루트 객체를 지정하면 배포 콘텐츠가 노출되지 않게 할 수 있습니다.

주제

- 기본 루트 객체를 지정하는 방법 (p. 215)
- 헤더가 기본 루트 객체에 대해 작동하는 방식 (p. 216)
- 루트 객체를 정의하지 않을 경우 CloudFront의 작동 방식 (p. 217)

기본 루트 객체를 지정하는 방법

웹 배포의 콘텐츠가 노출되는 것을 방지하거나 오류가 반환되는 것을 방지하려면 다음 단계를 수행하여 배포에 대한 기본 루트 객체를 지정합니다.

배포에 대한 기본 루트 객체를 지정하려면

1. 배포에서 가리키는 오리진에 기본 루트 객체를 업로드합니다.

   이 파일은 CloudFront에서 지원되는 모든 형식이 될 수 있습니다. 파일 이름에 대한 제한 사항 목록을 보려면 DistributionConfig 복합 형식단원에서 `DefaultRootObject` 요소의 설명을 참조하십시오.

   > **Note**
   >
   > 기본 루트 객체의 파일 이름이 너무 길거나 유효하지 않은 문자가 포함된 경우, CloudFront에서는 `HTTP 400 Bad Request - InvalidDefaultRootObject` 오류를 반환합니다. 또한 CloudFront에서는 5분 동안 코드를 캐싱하고 결과를 액세스 로그에 작성합니다.

2. 객체에 대한 권한이 CloudFront에 최소 `read` 액세스 권한을 부여하는지 확인합니다.

   Amazon S3 권한에 대한 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 액세스 제어 단원을 참조하십시오. Amazon S3 콘솔을 사용한 권한 업데이트에 대한 자세한 내용은 Amazon Simple Storage Service 콘솔 사용 설명서를 참조하십시오.

3. CloudFront 콘솔 또는 CloudFront API를 사용하여 기본 루트 객체를 참조하는 배포를 업데이트합니다.

   CloudFront 콘솔을 사용하여 기본 루트 객체를 지정하려면

   a. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
   b. 상단 창의 배포 목록에서 업데이트하려는 배포를 선택합니다.
   c. 배포 세부 정보 창의 일반 탭에서 편집을 선택합니다.
   d. Edit Distribution(배포 편집) 대화 상자에서 기본 루트 객체 필드에 기본 루트 객체의 파일 이름을 입력합니다.

      `index.html`과 같이 객체 이름만 입력하세요. 객체 이름 앞에 `/`를 추가하지 마십시오.
   e. 변경 사항을 저장하려면 예, 편집합니다를 선택합니다.

CloudFront API를 사용하여 구성을 업데이트하려면 배포에서 `DefaultRootObject` 요소의 값을 지정합니다. CloudFront API를 사용한 기본 루트 객체 지정에 대한 자세한 내용은 Amazon CloudFront API Reference의 PUT 배포 구성 단원을 참조하십시오.

4. 루트 URL을 요청하여 기본 루트 객체를 활성화했는지 확인합니다. 브라우저에 기본 루트 객체가 표시되지 않는 경우 다음 단계를 수행합니다.

   a. CloudFront 콘솔에서 배포의 상태를 확인하여 배포가 모두 완료되었는지 확인합니다.

   b. 2단계와 3단계를 반복하여 올바른 권한을 부여하고 기본 루트 객체를 지정하는 배포 구성을 올바르게 업데이트했는지 확인합니다.

헤더가 기본 루트 객체에 대해 작동하는 방식

여기서는 기본 루트 객체가 어떻게 작동하는지에 대한 예를 보여줍니다. 예를 들어 다음 요청은 `image.jpg` 객체를 가리킵니다.

```
http://d111111abcdef8.cloudfront.net/image.jpg
```

반면에, 다음 요청은 특정 객체 대신에 앞 예제와 동일한 배포의 루트 URL을 가리킵니다.

```
http://d111111abcdef8.cloudfront.net/
```

기본 루트 객체를 정의한 경우 배포의 루트를 호출하는 최종 사용자 요청은 기본 루트 객체를 반환합니다. 예를 들어, `index.html` 파일을 기본 루트 객체로 지정하는 경우, 그에 대한 요청은 다음과 같습니다.

```
http://d111111abcdef8.cloudfront.net/
```

반환 값은 다음과 같습니다.

```
http://d111111abcdef8.cloudfront.net/index.html
```

그러나, 기본 루트 객체를 정의한 경우 배포의 하위 디렉터리에 대한 최종 사용자 요청은 기본 루트 객체를 반환하지 않습니다. 예를 들어 `index.html`이 기본 루트 객체이고 CloudFront에서 CloudFront 배포 아래의 `install` 디렉터리에 대한 최종 사용자 요청을 받았다고 가정합니다.

```
http://d111111abcdef8.cloudfront.net/install/
```

`index.html`의 사본이 `install` 디렉터리에 나타나더라도 CloudFront에서는 기본 루트 객체를 반환하지 않습니다.

CloudFront에서 지원되는 모든 HTTP 메서드를 허용하도록 배포를 구성한 경우, 기본 루트 객체는 모든 메소드에 적용됩니다. 예를 들어 기본 루트 객체가 index.php이고 `POST` 요청을 도메인(http://example.com)의 루트에 제출하도록 애플리케이션을 작성한 경우, CloudFront에서는 요청을 http://example.com/index.php로 보냅니다.

CloudFront 기본 루트 객체의 동작은 Amazon S3 인덱스 문서의 동작과는 다릅니다. Amazon S3 버킷을 웹 사이트로 구성하고 인덱스 문서를 지정한 경우, 사용자가 버킷의 하위 디렉터리를 요청하더라도 Amazon S3에서는 인덱스 문서를 반환합니다. (인덱스 문서의 사본은 모든 하위 디렉터리에 나타나야 합니다.) Amazon S3 버킷을 웹 사이트로 구성하는 방법과 인덱스 문서에 대한 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 Amazon S3에서 웹 사이트 호스팅 장을 참조하십시오.

> **Important**
>
> 기본 루트 객체는 CloudFront 배포에만 적용한다는 점에 유의하십시오. 이 경우에도 오리진의 보안은 관리해야 합니다. 예를 들어 Amazon S3 오리진을 사용하는 경우, Amazon S3 버킷 ACL을 적절히 설정하여 버킷에 대해 원하는 액세스 수준을 보장해야 합니다.

루트 객체를 정의하지 않을 경우 CloudFront의 작동 방식

기본 루트 객체를 정의하지 않은 경우, 배포의 루트에 대한 요청은 오리진 서버로 전달됩니다. Amazon S3 오리진을 사용하는 경우 다음 중 하나가 반환될 수 있습니다.

- Amazon S3 버킷의 콘텐츠 목록 — 다음 조건 중 하나에 해당하는 경우 오리진의 콘텐츠가 CloudFront를 사용하여 배포에 액세스하는 사람에게 표시됩니다.
  - 버킷이 적절하게 구성되지 않았습니다.
  - 배포에 연결된 버킷과 이 버킷에 있는 객체에 대한 Amazon S3 권한은 모두에 대한 액세스 권한을 부여합니다.
  - 최종 사용자가 오리진 루트 URL을 사용하여 오리진에 액세스합니다.
- 오리진의 프라이빗 콘텐츠 목록 — 비공개 배포(사용자 및 CloudFront에만 액세스 권한이 있음)로 오리진을 구성하는 경우, 배포에 연결된 Amazon S3 버킷의 콘텐츠는 CloudFront를 통해 배포에 액세스하는 자격 증명을 가진 모두에게 표시됩니다. 이러한 경우 사용자는 오리진 루트 URL을 통해 콘텐츠에 액세스할 수 없습니다. 비공개 콘텐츠를 배포하는 방법에 대한 자세한 내용은 서명된 URL과 서명된 쿠키를 사용하여 프라이빗 콘텐츠 제공 (p. 115)를 참조하십시오.
- Error 403 Forbidden — 배포에 연결된 Amazon S3 버킷에 대한 권한이나 이 버킷에 있는 객체에 대한 권한이 CloudFront 및 모두에 대한 액세스를 거부하는 경우, CloudFront에서는 이 오류를 반환합니다.

문제 해결

Amazon CloudFront를 설정하여 콘텐츠를 배포하거나 Lambda@Edge를 사용할 때 발생할 수 있는 일반적인 문제를 해결하고 해결 방법을 찾아봅니다.

주제

- 배포 문제 해결 (p. 218)
- 오리진의 오류 응답 문제 해결 (p. 221)
- CloudFront 로드 테스트 (p. 228)

배포 문제 해결

이 문서의 정보를 사용하여 Amazon CloudFront 배포로 웹 사이트 또는 애플리케이션 설정 시 발생할 수 있는 인증서 오류, 액세스 거부 문제 또는 기타 일반적인 문제를 진단하고 해결할 수 있습니다.

주제

- 대체 도메인 이름을 추가하려고 하면 CloudFront에서 InvalidViewerCertificate 오류를 반환합니다 (p. 218)
- 배포에서 파일을 볼 수 없습니다. (p. 219)
- 오류 메시지: Certificate: <certificate-id> Is Being Used by CloudFront (p. 220)

대체 도메인 이름을 추가하려고 하면 CloudFront에서 InvalidViewerCertificate 오류를 반환합니다

배포에 대체 도메인 이름(CNAME)을 추가하려고 하면 CloudFront에서 `InvalidViewerCertificate` 오류를 반환하는 경우, 다음 정보를 검토하여 문제를 해결하십시오. 이 오류는 대체 도메인 이름을 추가하려면 먼저 다음 문제 중 하나를 해결해야 함을 뜻할 수 있습니다.

다음 오류는 CloudFront에서 대체 도메인 이름을 추가할 권한이 있는지 확인하는 순서로 나와 있습니다. 이는 CloudFront가 반환하는 오류를 기반으로 성공적으로 완료된 확인을 알 수 있으므로 문제를 해결하는 데 도움이 됩니다.

배포에 연결된 인증서가 없습니다.

> 대체 도메인 이름(CNAME)을 추가하려면 배포에 신뢰할 수 있는 유효한 인증서를 연결해야 합니다. 요구 사항을 검토하고 요구 사항을 충족하는 유효한 인증서를 얻고 배포에 연결한 후 다시 시도하십시오. 자세한 내용은 대체 도메인 이름 사용과 관련된 요구 사항 (p. 73) 단원을 참조하십시오.

연결한 인증서에 대한 인증서 체인에 너무 많은 인증서가 있습니다.

> 인증서 체인 하나에 최대 5개의 인증서만 포함할 수 있습니다. 체인의 인증서 수를 줄인 후 다시 시도하십시오.

인증서 체인에 현재 날짜에 유효하지 않은 인증서가 하나 이상 포함되어 있습니다.

> 추가한 인증서의 인증서 체인에 인증서가 아직 유효하지 않거나 인증서가 만료된 이유로 유효하지 않은 인증서가 하나 이상 있습니다. 인증서 체인의 인증서에 있는 Not Valid Before 및 Not Valid After 필드를 확인하여 모든 인증서가 나열한 날짜를 기반으로 유효한지 확인하십시오.

연결한 인증서가 신뢰할 수 있는 인증 기관(CA)에서 서명되지 않았습니다.

대체 도메인 이름을 확인하기 위해 CloudFront에 연결하는 인증서는 자체 서명된 인증서일 수 없습니다. 신뢰할 수 있는 CA에서 서명해야 합니다. 자세한 내용은 대체 도메인 이름 사용과 관련된 요구 사항 (p. 73) 단원을 참조하십시오.

연결한 인증서가 올바른 형식이 아닙니다.

인증서에 포함된 도메인 이름 및 IP 주소 형식과, 인증서 자체의 형식이 인증서 표준을 따라야 합니다.

CloudFront 내부 오류가 발생했습니다.

CloudFront가 내부 문제로 차단되어 인증서를 검사할 수 없습니다. 이 시나리오에서 CloudFront는 HTTP 500 상태 코드를 반환하고 인증서 연결과 관련된 내부 CloudFront 문제가 있음을 보여 줍니다. 몇 분 정도 기다린 후 인증서와 함께 대체 도메인 이름을 다시 추가해 보십시오.

연결한 인증서가 추가하려는 대체 도메인 이름을 포함하지 않습니다.

CloudFront는 사용자가 추가하는 대체 도메인 이름별로 해당 도메인 이름이 포함된 신뢰할 수 있는 인증 기관(CA)의 유효한 SSL/TLS 인증서를 연결하여 사용 권한을 검증할 것을 요구합니다. 추가하려는 CNAME이 포함된 도메인 이름을 포함하도록 인증서를 업데이트하십시오. 와일드카드가 있는 도메인 이름 사용에 대한 자세한 내용과 예는 대체 도메인 이름 사용과 관련된 요구 사항 (p. 73) 단원을 참조하십시오.

배포에서 파일을 볼 수 없습니다.

CloudFront 배포에서 파일을 볼 수 없는 경우, 다음 주제에서 일반적인 해결 방법을 참조하십시오.

CloudFront 및 Amazon S3 모두에 가입했습니까?

Amazon CloudFront에서 Amazon S3 오리진을 사용하려면 CloudFront 및 Amazon S3에 각각 별도로 가입해야 합니다. CloudFront 및 Amazon S3 가입에 대한 자세한 내용은 Amazon CloudFront 설정 (p. 10) 단원을 참조하십시오.

Amazon S3 버킷과 객체 권한을 올바르게 설정했습니까?

Amazon S3 오리진에서 CloudFront를 사용하는 경우 콘텐츠의 원래 버전은 S3 버킷에 저장됩니다. CloudFront에서 Amazon S3을 사용하는 가장 쉬운 방법은 모든 객체를 Amazon S3에서 공개적으로 읽을 수 있도록 설정하는 것입니다. 이렇게 하려면 Amazon S3에 업로드하는 각 객체에 대해 퍼블릭 읽기 권한을 명시적으로 활성화해야 합니다.

콘텐츠를 공개적으로 사용할 수 없으면 CloudFront에서 액세스할 수 있도록 CloudFront 오리진 액세스 ID(OAI)를 만들어야 합니다. CloudFront 원본 액세스 ID에 대한 자세한 내용은 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오.

객체 속성과 버킷 속성은 독립적입니다. Amazon S3의 각 객체에 권한을 명시적으로 부여해야 합니다. 객체는 버킷에서 속성을 상속하지 않으므로 객체 속성을 버킷과 별도로 설정해야 합니다.

대체 도메인 이름(CNAME)을 올바르게 구성했습니까?

도메인 이름에 대한 기존 CNAME 레코드가 이미 있는 경우 해당 레코드를 업데이트하거나 배포의 도메인 이름을 가리키는 새 리소스 레코드 세트로 바꿉니다.

또한 CNAME 레코드가 Amazon S3 버킷이 아니라 배포의 도메인 이름을 가리키는지 확인하십시오. DNS 시스템의 CNAME 레코드가 배포의 도메인 이름을 가리키는지 확인할 수 있습니다. 이렇게 하려면 Dig와 같은 DNS 도구를 사용합니다. dig에 대한 자세한 내용은 http://www.kloth.net/services/dig.php를 참조하십시오.

다음 예는 `images.example.com`이라고 하는 도메인 이름에 대한 dig 요청과 관련 응답 부분을 보여 줍니다. `ANSWER SECTION` 아래에서 `CNAME`를 포함하는 줄을 참조하십시오. CNAME 오른쪽에 있는 값이 CloudFront 배포의 도메인 이름이면 도메인 이름의 CNAME 레코드가 올바르게 설정된 것입니다. 이 값이 Amazon S3 오리진 서버 버킷이나 다른 도메인 이름이면 CNAME 레코드가 잘못 설정된 것입니다.

```
[prompt]>
dig images.example.com

; <<> DiG 9.3.3rc2 <<> images.example.com
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15917
;; flags: qr rd ra; QUERY: 1, ANSWER: 9, AUTHORITY: 2, ADDITIONAL: 0
;; QUESTION SECTION:
;images.example.com.   IN  A
;; ANSWER SECTION:
images.example.com. 10800 IN CNAME d111111abcdef8.cloudfront.net.
...
...
```

CNAME에 대한 자세한 내용은 대체 도메인 이름(CNAME)을 추가하여 파일에 대해 사용자 지정 URL 사용 (p. 66)을 참조하십시오.

CloudFront 배포에 대한 올바른 URL을 참조하고 있습니까?

참조하고 있는 URL에 CloudFront 버킷 또는 사용자 지정 오리진이 아닌 Amazon S3 배포 도메인 이름(또는 CNAME)이 사용되는지 확인하십시오.

사용자 지정 오리진 문제 해결과 관련하여 도움이 필요합니까?

사용자 지정 오리진 문제를 해결하는 데 AWS의 도움이 필요한 경우 아마도 요청의 `X-Amz-Cf-Id` 헤더 항목을 검사해야 할 것입니다. 이러한 항목을 로그에 기록하고 있지 않은 경우 향후 참조를 위해 로그에 기록할 수 있습니다. 자세한 내용은 Amazon EC2 또는 기타 사용자 지정 오리진 사용 (p. 63) 단원을 참조하십시오. 자세한 도움말은 AWS Support Center를 참조하십시오.

RTMP 배포에서 파일을 볼 수 없습니다.

RTMP 배포에서 파일을 볼 수 없는 경우 URL과 재생 콘텐츠가 올바르게 구성되었습니까? RTMP 배포에서는 HTTP 대신에 RTMP 프로토콜을 사용하며 재생 클라이언트에 대한 구성을 조금 변경해야 합니다. RTMP 배포 만들기에 대한 자세한 내용은 RTMP를 사용한 미디어 파일 스트리밍용 작업 목록 (p. 271)을 참조하십시오.

오류 메시지: Certificate: <certificate-id> Is Being Used by CloudFront

문제: IAM 인증서 스토어에서 SSL/TLS를 삭제하려고 할 때 "Certificate: <certificate-id> is being used by CloudFront." 메시지가 표시됩니다.

해결 방법: 모든 CloudFront 웹 배포를 기본 CloudFront 인증서 또는 사용자 지정 SSL/TLS 인증서와 연결해야 합니다. SSL/TLS 인증서를 삭제하기 전에 이 인증서를 교체하거나(현재 사용자 지정 SSL/TLS 인증서를 다른 사용자 지정 SSL/TLS 인증서로 바꾸기) 사용자 지정 SSL/TLS 인증서 사용을 기본 CloudFront 인증서 사용으로 되돌려야 합니다. 이를 해결하려면 다음 절차 중 하나의 단계를 수행합니다.

- SSL/TLS 인증서 회전 (p. 113)

- 사용자 지정 SSL/TLS 인증서에서 기본 CloudFront 인증서로 되돌리기 (p. 114)

오리진의 오류 응답 문제 해결

CloudFront가 오리진에 객체를 요청했는데 오리진에서 HTTP 4xx 또는 5xx 상태 코드를 반환하는 경우, CloudFront와 오리진 사이에 통신 문제가 있는 것입니다. 다음 주제에서는 이 HTTP 상태 코드 중 일부의 일반적인 원인과 가능한 해결 방법 몇 가지를 설명합니다.

주제

- HTTP 400 상태 코드(잘못된 요청) (p. 221)
- HTTP 500 상태 코드(Lambda 실행 오류) (p. 222)
- HTTP 502 상태 코드(잘못된 게이트웨이) (p. 222)
- HTTP 502 상태 코드(Lambda 확인 오류) (p. 224)
- HTTP 503 상태 코드(Lambda 제한 초과됨) (p. 225)
- HTTP 503 상태 코드(Service Unavailable) (p. 225)
- HTTP 504 상태 코드(게이트웨이 제한 시간) (p. 226)

HTTP 400 상태 코드(잘못된 요청)

CloudFront 배포가 HTTP 상태 코드 400 잘못된 요청 및 다음과 유사한 메시지와 함께 오류 응답을 보낼 수 있습니다.

권한 부여 헤더 형식이 잘못되었습니다. 리전 '*<AWS ##>*'이 잘못되었습니다. '*<AWS ##>*'이어야 합니다.

예:

권한 부여 헤더 형식이 잘못되었습니다. 리전 'us-east-1'이 잘못되었습니다. 'us-west-2'여야 합니다.

다음과 같은 시나리오에서 이 문제가 발생할 수 있습니다.

1. CloudFront 배포의 오리진은 Amazon S3 버킷입니다.
2. S3 버킷을 특정 AWS 리전에서 다른 AWS 리전으로 이동했습니다. 즉, S3 버킷을 삭제한 후 나중에 동일한 버킷 이름을 가진 새 버킷을 만들었지만 원래 S3 버킷이 있던 AWS 리전과 다릅니다.

이 오류를 수정하려면 버킷의 현재 AWS 리전에서 S3 버킷을 찾도록 CloudFront 배포를 업데이트하십시오.

CloudFront 배포를 업데이트하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 이 오류를 일으키는 배포를 선택합니다.
3. Origins and Origin Groups(오리진 및 오리진 그룹)를 선택합니다.
4. 이동한 S3 버킷의 오리진을 찾습니다. 이 오리진 옆의 확인란을 선택한 후 편집을 선택합니다.
5. 예, 편집합니다를 선택합니다. Yes, Edit(예, 편집합니다.)을 선택하기 전 설정을 변경할 필요가 없습니다.

이 단계를 끝내면 CloudFront가 배포를 다시 배포합니다. 배포가 진행되는 동안 CloudFront 콘솔에서 배포 상태가 진행 중으로 바뀝니다. 배포가 완료되면 배포 상태가 배포 완료로 바뀝니다. 그러면 AuthorizationHeaderMalformed 오류 응답 수신을 중지해야 합니다. 상태가 배포 완료로 변경된 후에도 이 오류 수신을 중지하기까지 시간이 좀 걸릴 수 있습니다.

HTTP 500 상태 코드(Lambda 실행 오류)

Lambda@Edge를 사용하는 경우 HTTP 500 상태 코드는 Lambda 함수가 실행 오류를 반환했음을 나타낼 수 있습니다. Lambda@Edge 오류 문제 해결에 대한 자세한 내용은 Lambda@Edge 함수 테스트 및 디버깅 (p. 306) 단원을 참조하십시오.

HTTP 502 상태 코드(잘못된 게이트웨이)

HTTP 502 상태 코드(잘못된 게이트웨이)는 CloudFront가 오리진 서버에 연결할 수 없어 요청된 객체를 제공할 수 없다는 뜻입니다.

주제

- CloudFront와 사용자 지정 오리진 서버 간의 SSL/TLS 협상 실패 (p. 222)
- 오리진이 지원되는 암호화/프로토콜에 응답하지 않음 (p. 223)
- 오리진의 SSL/TLS 인증서가 만료되었거나 잘못되었거나 자체 서명되었거나, 인증서 체인의 순서가 잘못됨 (p. 223)
- 오리진이 오리진 설정의 지정된 포트에서 응답하지 않음 (p. 224)
- CloudFront Was Not Able to Resolve Your Origin Domain Due to DNS Issues (p. 224)
- Lambda@Edge 함수 유효성 검사 오류 (p. 224)

CloudFront와 사용자 지정 오리진 서버 간의 SSL/TLS 협상 실패

사용자 지정 오리진을 사용 중이며 CloudFront와 오리진 사이에서 HTTPS를 요구하도록 CloudFront를 구성한 경우, 도메인 이름 불일치가 문제가 될 수 있습니다. 오리진에 설치된 SSL/TLS 인증서에는 일반 이름 필드에 도메인 이름이 포함되며 Subject Alternative Names(주제 대체 이름) 필드에도 몇 번 더 포함될 수 있습니다. CloudFront는 인증서 도메인 이름 내 와일드카드 문자 사용을 지원합니다. 인증서의 도메인 이름 중 하나가 다음 값 모두와 또는 하나와 일치해야 합니다.

- 배포의 해당 오리진의 Origin Domain Name(오리진 도메인 이름)에 대해 지정된 값입니다.
- `Host` 헤더를 오리진으로 전달하도록 CloudFront를 구성한 경우 `Host` 헤더의 값입니다. `Host` 헤더를 오리진으로 전달하는 방법에 대한 자세한 내용은 요청 헤더에 따라 콘텐츠 캐싱 (p. 200) 단원을 참조하십시오.

도메인 이름이 일치하지 않으면 SSL/TLS 핸드쉐이크가 실패하고, CloudFront는 HTTP 상태 코드 502(잘못된 게이트웨이)를 반환하고 `X-Cache` 헤더를 `Error from cloudfront`로 설정합니다.

온라인 SSL 확인 프로그램 또는 OpenSSL을 사용하여 인증서의 도메인 이름이 `Host` 헤더 또는 배포의 Origin Domain Name(오리진 도메인 이름)과 일치하는지 여부를 파악할 수 있습니다. 도메인 이름이 일치하지 않는 경우, 다음 두 가지 옵션이 있습니다.

- 해당하는 도메인 이름이 포함된 새 SSL/TLS 인증서를 받습니다.

  AWS Certificate Manager(ACM)을 사용 중인 경우 새 인증서를 요청하는 방법은 AWS Certificate Manager 사용 설명서의 인증서 요청 단원을 참조하십시오.
- CloudFront가 더 이상 SSL을 사용하여 오리진 연결을 시도하지 않도록 배포 구성을 변경합니다.

온라인 SSL 확인 프로그램

SSL 테스트 도구를 찾으려면 인터넷에서 "online ssl checker"를 검색합니다. 일반적으로, 도메인 이름을 지정하면 SSL/TLS 인증서에 대한 다양한 정보가 도구에 표시됩니다. 인증서의 일반 이름 또는 Subject Alternative Names(주제 대체 이름) 필드에 도메인 이름이 나와 있는지 확인합니다.

OpenSSL

CloudFront가 오리진과 연결할 수 있는지 여부를 알아보기 위해, OpenSSL을 사용하여 오리진과 SSL/TLS 연결을 해 보고 오리진의 인증서가 올바르게 구성되었는지 확인할 수 있습니다. OpenSSL로 연결을 할 수 있으면 오리진 서버의 인증서에 대한 정보가 표시됩니다.

사용하는 명령은 SNI(Server Name Indication)를 지원하는 클라이언트를 사용하는지 여부에 따라 달라집니다.

SNI를 지원하는 클라이언트

```
openssl s_client -connect domainname:443 -servername domainname
```

SNI를 지원하지 않는 클라이언트

```
openssl s_client -connect domainname:443
```

*domainname*을 해당하는 값으로 바꿉니다.

- **`Host`** 헤더를 오리진으로 전달하지 않는 경우 – *domainname*을 오리진의 도메인 이름으로 바꿉니다.
- Host 헤더를 오리진으로 전달하는 경우 – *domainname*을 CloudFront 배포에 사용 중인 CNAME으로 바꿉니다.

오리진이 지원되는 암호화/프로토콜에 응답하지 않음

CloudFront는 암호화 및 프로토콜을 사용하여 오리진 서버에 연결합니다. CloudFront에서 지원하는 암호화 및 프로토콜 목록은 지원되는 암호화 및 프로토콜 (p. 98)을 참조하십시오. 오리진이 SSL/TLS 교환에서 이러한 암호화 또는 프로토콜 중 하나에 응답하지 않는 경우, CloudFront가 연결되지 않은 것입니다. SSL Labs를 사용하여 오리진에서 암호화 및 프로토콜을 지원하는지 확인할 수 있습니다.

- SSL Labs

  호스트 이름 필드에 오리진의 도메인 이름을 입력한 다음 제출을 선택합니다. 테스트에서 일반 이름 및 대체 이름 필드를 검토하여 오리진의 도메인 이름과 일치하는지 확인합니다.

  테스트가 완료되면 테스트 결과에서 프로토콜 및 암호 글부 섹션을 찾아 오리진에서 암호화 또는 프로토콜을 지원하는지 확인합니다. 이를 지원되는 암호화 및 프로토콜 (p. 98) 목록과 비교합니다.

  > **Note**
  >
  > Elastic Load Balancing를 사용 중인 경우, 암호화 및 프로토콜 설정 방법을 알아보려면 Elastic Load Balancing 사용 설명서의 Elastic Load Balancing의 SSL 보안 정책 단원을 참조하십시오. 미리 정의된 보안 정책 ELBSecurityPolicy-2016-08을 사용하면 엘라스틱 로드 밸런서에 대한 CloudFrontaccess를 제공합니다. 사용자 지정 정책을 사용하여 이를 추가로 제한하려면 CloudFront에서 지원하는 암호화를 허용해야 합니다.

오리진의 SSL/TLS 인증서가 만료되었거나 잘못되었거나 자체 서명되었거나, 인증서 체인의 순서가 잘못됨

오리진 서버에서 다음을 반환하면, CloudFront는 TCP 연결을 끊고 HTTP 상태 코드 502(잘못된 게이트웨이)를 반환하며, `X-Cache` 헤더를 `Error from cloudfront`로 설정합니다.

- 만료된 인증서
- 잘못된 인증서
- 자체 서명된 인증서

- 인증서 체인의 순서가 잘못됨

  Note

  중간 인증서를 포함하여 인증서의 전체 체인이 없는 경우, CloudFront는 TCP 연결을 끊습니다.

사용자 지정 오리진 서버에 SSL/TLS 인증서를 설치하는 것에 대한 자세한 내용은 CloudFront와 사용자 지정 오리진 간의 통신에 HTTPS 요구 (p. 94)를 참조하십시오.

오리진이 오리진 설정의 지정된 포트에서 응답하지 않음

CloudFront 배포에 오리진을 만드는 경우, CloudFront가 HTTP 및 HTTPS 트래픽을 통해 오리진에 연결하는 포트를 설정할 수 있습니다. 기본적으로 TCP 80/443이 있습니다. 이러한 포트를 수정할 수 있습니다. 오리진이 어떤 이유로 이러한 포트의 트래픽을 거부하거나 백엔드 서버가 포트에서 응답하지 않는 경우, CloudFront가 연결되지 않습니다.

이러한 문제를 해결하려면 인프라에서 실행 중인 방화벽을 확인하고, 방화벽이 지원되는 IP 범위를 차단하지 않는지 확인하십시오. 자세한 내용은 Amazon Web Services 일반 참조의 AWS IP 주소 범위 단원을 참조하십시오. 또한 웹 서버가 오리진에서 실행되는지 확인하십시오.

CloudFront Was Not Able to Resolve Your Origin Domain Due to DNS Issues

만료되었거나 캐시에 저장되지 않은 객체에 대한 요청이 들어오면 CloudFront는 업데이트된 객체를 가져오기 위해 오리진에 요청합니다. 성공적으로 오리진에 요청하기 위해 CloudFront는 오리진 도메인 이름에 대해 DNS 확인을 수행합니다. 하지만 도메인을 호스트하는 DNS 서비스에 문제가 있는 경우 CloudFront가 도메인 이름을 확인하여 IP 주소를 가져올 수 없어 502 오류가 발생합니다. 이 문제를 해결하려면 DNS 공급자에 문의하십시오. 또는 Amazon Route 53을 사용하는 경우 Amazon Route 53 DNS를 참조하십시오.

이 문제를 추가로 해결하려면 오리진 루트 도메인 또는 zone apex(예: `example.com`)의 권한 있는 이름 서버가 올바르게 작동 중인지 확인하십시오. 그러면 권한 있는 이름 서버가 요청을 수신하며 도메인과 연결되어 있고 CloudFront 배포를 설정하는 데 사용된 DNS 서버와 동일한 IP 주소를 반환합니다. 다음 명령을 사용하여 apex 오리진의 이름 서버를 찾습니다.

```
dig OriginAPEXDomainName NS +short
nslookup -query=NS OriginAPEXDomainName
```

이름 서버의 이름을 아는 경우 다음 명령을 사용하여 해당 이름에 대해 오리진의 도메인 이름을 쿼리하여 각각 다음 답변으로 응답하는지 확인합니다.

```
dig OriginDomainName @NameServerFromAbove
nslookup OriginDomainName NameServerFromAbove
```

Lambda@Edge 함수 유효성 검사 오류

Lambda@Edge를 사용하는 경우 HTTP 502 상태 코드는 Lambda 함수 응답이 잘못 구성되어 있거나 잘못된 내용을 포함하고 있음을 나타낼 수 있습니다. Lambda@Edge 오류 문제 해결에 대한 자세한 내용은 Lambda@Edge 함수 테스트 및 디버깅 (p. 306) 단원을 참조하십시오.

HTTP 502 상태 코드(Lambda 확인 오류)

Lambda@Edge를 사용하는 경우 HTTP 502 상태 코드는 Lambda 함수 응답이 잘못 구성되어 있거나 잘못된 내용을 포함하고 있음을 나타낼 수 있습니다. Lambda@Edge 오류 문제 해결에 대한 자세한 내용은 Lambda@Edge 함수 테스트 및 디버깅 (p. 306) 단원을 참조하십시오.

HTTP 503 상태 코드(Lambda 제한 초과됨)

Lambda@Edge를 사용하는 경우 HTTP 503 상태 코드는 Lambda 서비스가 오류를 반환했음을 나타낼 수 있습니다. 오류는 다음 중 하나로 인해 발생할 수 있습니다.

- 함수 실행 수가 AWS 리전에서 실행을 조절(동시 실행 또는 호출 빈도)하기 위해 Lambda가 설정한 할당량(이전에는 제한이라고 함) 중 하나를 초과했습니다.
- 함수가 Lambda 함수의 제한 시간 할당량을 초과했습니다.

AWS Lambda 할당량에 대한 자세한 내용은 AWS Lambda Developer Guide의 AWS Lambda 제한을 참조하십시오. Lambda@Edge 오류 문제 해결에 대한 자세한 내용은 Lambda@Edge 함수 테스트 및 디버깅 (p. 306) 단원을 참조하십시오.

HTTP 503 상태 코드(Service Unavailable)

HTTP 503 상태 코드(Service Unavailable)는 일반적으로 오리진 서버의 성능 문제를 나타냅니다. 드물지만 이 상태 코드가 엣지 로케이션의 리소스 제한 때문에 CloudFront가 일시적으로 요청을 충족할 수 없음을 나타냅니다.

주제

- Origin Server Does Not Have Enough Capacity to Support the Request Rate (p. 225)
- CloudFront Caused the Error Due to Resource Constraints at the Edge Location (p. 225)

Origin Server Does Not Have Enough Capacity to Support the Request Rate

오리진 서버가 수신 요청으로 가득 차는 경우 CloudFront가 이 오류를 생성합니다. 그런 다음 CloudFront가 다시 사용자에게 오류를 전달합니다. 이 문제를 해결하려면 다음 솔루션을 시도해 보십시오.

- Amazon S3를 오리진 서버로 사용하는 경우 키 이름 지정 모범 사례를 따라 Amazon S3의 성능을 최적화합니다. 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 요청 빈도 및 성능 고려 사항 단원을 참조하십시오.
- Elastic Load Balancing를 오리진 서버로 사용하는 경우 503 Error Classic을 참조하십시오.
- 사용자 지정 오리진을 사용하는 경우, 애플리케이션 로그를 검사하여 오리진에 메모리, CPU, 디스크 크기 등의 리소스가 충분한지 확인합니다. Amazon EC2를 백엔드로 사용하는 경우 인스턴스 유형에 수신 요청을 충족하는 적절한 리소스가 있는지 확인합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 인스턴스 유형 단원을 참조하십시오.

CloudFront Caused the Error Due to Resource Constraints at the Edge Location

드문 경우지만 CloudFront가 사용 가능한 차선의 엣지 로케이션으로 요청을 라우팅할 수 없어서 요청을 충족할 수 없는 경우에 이 오류가 발생합니다. 이 오류는 CloudFront 배포에 대한 부하 테스트를 수행할 때 흔히 발생합니다. 이러한 오류를 방지하려면 503(용량 초과) 오류 방지를 위한 CloudFront 로드 테스트 (p. 228) 지침을 따르십시오.

프로덕션 환경에서 이러한 오류가 발생하면 AWS 지원 팀에 문의하십시오.

HTTP 504 상태 코드(게이트웨이 제한 시간)

HTTP 504 상태 코드(게이트웨이 제한 시간)는 CloudFront가 오리진에 요청을 전달했을 때(요청된 객체가 엣지 캐시에 없었기 때문에) 다음 중 하나가 발생했음을 나타냅니다.

- 오리진이 CloudFront에 HTTP 504 상태 코드를 반환했습니다.
- 요청이 만료되기 전에 오리진이 응답하지 않았습니다.

방화벽 또는 보안 그룹에 의해 오리진에 대해 트래픽이 차단된 경우나 오리진이 인터넷에서 액세스가 불가능한 경우에 CloudFront는 HTTP 504 상태 코드를 반환합니다. 이러한 문제들이 있는지 먼저 확인합니다. 그런 다음, 액세스 문제가 아닌 경우에는 애플리케이션 지연 및 서버 제한 시간을 탐색하여 문제를 확인하고 시정합니다.

주제

- CloudFront 트래픽을 허용하도록 오리진 서버에서 방화벽 구성 (p. 226)
- CloudFront 트래픽을 허용하도록 오리진 서버에서 보안 그룹 구성 (p. 226)
- 사용자 지정 오리진 서버를 인터넷에서 액세스할 수 있도록 설정 (p. 226)
- 오리진 서버의 애플리케이션에서 지연 응답을 찾아서 수정 (p. 227)

CloudFront 트래픽을 허용하도록 오리진 서버에서 방화벽 구성

오리진 서버의 방화벽에 의해 CloudFront 트래픽이 차단되는 경우에 CloudFront는 HTTP 504 상태 코드를 반환하는데, 다른 문제를 확인하기 전에 먼저 이 문제가 아닌지 확인하는 것이 좋습니다.

이것이 방화벽 문제인지를 판단하기 위해 사용하는 방법은 오리진 서버가 사용하는 시스템에 따라 다릅니다.

- Linux 서버에서 IPTable 방화벽을 사용하는 경우에는 IPTable에서의 작업에 도움이 되는 도구 및 정보를 검색할 수 있습니다.
- Windows 서버에서 Windows 방화벽을 사용하는 경우에는 Microsoft Technet의 방화벽 규칙 추가 또는 편집을 참조하십시오.

오리진 서버에서 방화벽 구성을 평가할 때는 게시된 IP 주소 범위를 토대로 CloudFront 엣지 로케이션에서의 트래픽을 차단하는 방화벽이나 보안 규칙이 있는지 찾아봅니다.

CloudFront IP 주소 범위가 오리진 서버의 화이트리스트에 등록되어 있으면 변경 사항이 포함되도록 서버의 보안 규칙을 업데이트합니다. Amazon Simple Notification Service(SNS) 주제를 구독하면 IP 주소 범위 파일이 업데이트될 때 알림을 수신할 수 있습니다. 알림을 수신한 후에는 코드를 사용하여 파일을 찾아서 구문 분석하고 로컬 환경에 맞게 조정할 수 있습니다. 자세한 내용은 Amazon SNS를 통한 AWS 퍼블릭 IP 주소 변경 사항 구독을 참조하십시오.

CloudFront 트래픽을 허용하도록 오리진 서버에서 보안 그룹 구성

오리진이 Elastic Load Balancing을 사용하는 경우에는 ELB 보안 그룹을 검토해서 보안 그룹이 CloudFront에서의 인바운드 트래픽을 허용하는지 확인합니다.

또한 AWS Lambda을 사용하여 보안 그룹을 자동 업데이트함으로써 CloudFront에서의 인바운드 트래픽을 허용할 수도 있습니다.

사용자 지정 오리진 서버를 인터넷에서 액세스할 수 있도록 설정

인터넷에서 공개적으로 사용할 수 없다는 점에서 CloudFront가 사용자 지정 오리진 서버를 액세스할 수 없는 경우, CloudFront는 HTTP 504 오류를 반환합니다.

CloudFront 엣지 로케이션은 인터넷을 통해 오리진 서버에 연결됩니다. 사용자 지정 오리진이 사설 네트워크에 있는 경우에는 CloudFront가 이를 액세스할 수 없습니다. 따라서 내부 Classic Load Balancer를 포함한 프라이빗 서버를 CloudFront에서 오리진 서버로 사용할 수 없습니다.

인터넷 트래픽을 오리진 서버에 연결할 수 있는지 확인하려면 다음 명령(OriginDomainName은 서버의 도메인 이름)을 실행하십시오.

HTTPS 트래픽의 경우:

- nc -zv OriginDomainName 443
- telnet OriginDomainName 443

HTTP 트래픽의 경우:

- nc -zv OriginDomainName 80
- telnet OriginDomainName 80

오리진 서버의 애플리케이션에서 지연 응답을 찾아서 수정

애플리케이션에서 응답에 너무 많은 시간이 소요되거나 제한 시간 값이 너무 낮게 설정된 결과로 인해 서버 제한 시간이 발생하는 경우가 종종 있습니다.

HTTP 504 오류를 방지하기 위한 일시적인 해결책으로 배포 시 CloudFront 제한 시간 값을 더 높게 설정하는 방법이 있기는 합니다. 그러나 애플리케이션 및 오리진 서버에서 성능 및 지연 문제가 해결되었는지 먼저 확인하는 것이 좋습니다. 제한 시간 값을 적당하게 설정하면 HTTP 504 오류를 방지하고 사용자에게 뛰어난 응답성을 제공할 수 있습니다.

여기에는 성능 문제를 찾아서 해결할 때 거치게 되는 단계들이 개략적으로 나와 있습니다.

1. 정상 부하 및 고부하에서 웹 애플리케이션의 지연 시간(응답성)을 측정합니다.
2. 필요할 경우 CPU나 메모리 같은 리소스를 추가합니다. 고부하 상황을 지원하도록 데이터베이스 쿼리를 튜닝하는 등 문제 해결을 위한 기타 조치를 취합니다.
3. 필요할 경우 CloudFront 웹 배포의 제한 시간 값을 조정합니다.

아래에는 각 단계에 대한 세부 정보가 나와 있습니다.

정상 부하 및 고부하에서 지연 시간 측정

하나 이상의 백엔드 웹 애플리케이션 서버가 고부하를 경험하고 있는지 판단하려면 각 서버에서 아래의 Linux curl 명령을 실행합니다.

```
curl -w "Connect time: %{time_connect} Time to first byte:
%{time_starttransfer} Total time: %{time_total} \n" -o /dev/null https://
www.example.com/yourobject
```

> **Note**
>
> 서버에서 Windows를 실행하는 경우에는 유사한 명령을 실행하도록 Windows용 cURL을 검색 및 다운로드할 수 있습니다.

서버에서 실행되는 애플리케이션의 지연 시간을 측정 및 평가할 때는 다음 사항에 유의하십시오.

- 지연 값은 각 애플리케이션에 상대적입니다. 하지만 초 단위가 아니라 밀리초 단위의 TTFB(Time To First Byte)가 적당합니다.

- 정상 부하에서 애플리케이션 지연 시간을 측정한 결과 문제가 없었더라도 고부하에서는 최종 사용자가 제한 시간을 경험할 수 있다는 점에 유의하십시오. 수요가 높으면 서버에서 응답이 지연되거나 아예 응답이 없을 수 있습니다. 고부하 지연 문제를 방지하려면 CPU, 메모리, 디스크 읽기/쓰기 같은 서버의 리소스를 확인하여 고부하에 맞게 서버 용량을 확장합니다.

  다음 Linux 명령을 실행하여 Apache 프로세스에서 사용되는 메모리를 확인할 수 있습니다.

  ```
  watch -n 1 "echo -n 'Apache Processes: ' && ps -C apache2 --no-headers | wc -
  l && free -m"
  ```
- 서버에서 CPU 활용도가 높으면 애플리케이션 성능이 크게 낮아질 수 있습니다. 백엔드 서버에서 Amazon EC2 인스턴스를 사용하는 경우에는 서버의 CloudWatch 지표를 검토하여 CPU 활용도를 확인합니다. 자세한 내용은 Amazon CloudWatch 사용 설명서를 참조하십시오. 자체 서버를 사용 중인 경우에는 서버 도움말 문서에서 CPU 활용도를 확인할 수 있는 방법에 대한 지침을 참조하십시오.
- 요청의 수가 많을 때 데이터베이스 쿼리의 실행 속도가 느려지는 것과 같이 고부하 상황에서 발생할 수 있는 기타 문제들을 확인하십시오.

리소스를 추가하고 서버 및 데이터베이스 튜닝

애플리케이션 및 서버의 응답성을 평가한 후에는 정상 부하 및 고부하 상황을 위해 충분한 리소스가 구축되어 있는지 확인합니다.

- 자체 서버를 가지고 있는 경우에는 평가 결과를 토대로 CPU, 메모리 및 디스크 공간이 최종 사용자 요청을 처리하기에 충분한지 확인합니다.
- Amazon EC2 인스턴스를 백엔드 서버로 사용하는 경우 인스턴스 유형에 수신 요청을 이행하기에 적절한 리소스가 있는지 확인합니다. 자세한 내용은 Amazon EC2 사용 설명서의 인스턴스 유형 단원을 참조하십시오.

뿐만 아니라, 제한 시간을 방지하려면 다음과 같은 튜닝 조치를 고려하십시오.

- cURL 명령이 반환한 TTFB(Time To First Byte) 값이 높은 경우에는 애플리케이션의 성능을 개선하기 위한 조치를 취합니다. 애플리케이션 응답성을 높이면 제한 시간 오류가 줄어듭니다.
- 성능 저하 없이 많은 양의 요청을 처리할 수 있도록 데이터베이스 쿼리를 튜닝합니다.
- 백엔드 서버에서 연결 유지(영구) 연결을 설정합니다. 이 옵션은 후속 요청이나 사용자에 대해 연결을 다시 설정해야 하는 경우에 발생하는 지연을 방지하는 데 도움이 됩니다.
- ELB를 오리진으로 사용하는 경우에는 지식 센터의 ELB의 긴 지연 시간을 해결하는 방법 기사에 나오는 제안을 검토하여 지연 시간을 줄일 수 있는 방법을 알아보십시오.

필요할 경우 CloudFront 제한 시간 값을 조정합니다.

애플리케이션 성능 저하, 오리진 서버 용량 및 기타 문제를 평가하고 해결했지만 최종 사용자가 여전히 HTTP 504 오류를 경험하고 있는 경우에는 오리진 응답 제한 시간에 대해 웹 배포에 지정되어 있는 시간을 변경하는 것을 반드시 고려해봐야 합니다. 자세한 내용은 오리진 응답 제한 시간 단원을 참조하십시오.

CloudFront 로드 테스트

기존 로드 테스트 방식은 CloudFront에서 DNS를 사용하여 지리적으로 분산된 엣지 로케이션으로 각 엣지 로케이션 내에 로드를 분배하기 때문에 CloudFront에는 잘 맞지 않았습니다. 클라이언트가 CloudFront에서 콘텐츠를 요청하면 이 클라이언트는 IP 주소 집합이 포함된 DNS 응답을 받습니다. DNS를 반환하는 IP 주소 중 하나에만 요청을 전송하여 테스트하는 경우 단일 CloudFront 엣지 로케이션에서 리소스의 소규모 하위 집합만 테스트하게 되며, 이는 실제 트래픽 패턴을 정확하게 나타내지 않습니다. 요청된 데이터 볼륨에 따라, 이 방법을 통한 테스팅은 CloudFront 서버의 해당 소규모 하위 집합을 오버로드하고 성능을 떨어뜨릴 수 있습니다.

CloudFront는 여러 지리적 리전에서 서로 다른 클라이언트 IP 주소와 DNS 확인 프로그램을 보유한 최종 사용자를 위해 규모를 조정하도록 설계되어 있습니다. CloudFront 성능을 정확하게 평가하는 로드 테스팅을 수행하려면 다음 사항을 모두 수행하는 것이 좋습니다.

- 여러 지리적 리전에서 클라이언트 요청을 보냅니다.
- 각 클라이언트에서 독립적인 DNS 요청을 만들도록 테스트를 구성합니다. 그러면 클라이언트별로 DNS에서 서로 다른 IP 주소 집합을 수신하게 됩니다.
- 요청을 수행하는 각 클라이언트의 경우 DNS에서 반환되는 IP 주소 집합 전반에 클라이언트 요청을 분산합니다. 이렇게 하면 CloudFront 엣지 로케이션의 여러 서버에 로드가 분산될 수 있습니다.

요청 및 응답 동작

다음 단원에서는 CloudFront에서 최종 사용자 요청을 처리하고 Amazon S3 또는 사용자 지정 오리진에 이 요청을 전달하는 방법과, CloudFront에서 4xx 및 5xx HTTP 상태 코드를 처리하고 캐싱하는 방법을 비롯해 CloudFront에서 오리진의 응답을 처리하는 방법을 설명합니다.

주제

- Amazon S3 오리진에 대한 요청 및 응답 동작 (p. 230)
- 사용자 지정 오리진에 대한 요청 및 응답 동작 (p. 236)
- 오리진 그룹에 대한 요청 및 응답 동작 (p. 247)
- 사용자 지정 헤더를 오리진 요청에 추가 (p. 248)
- CloudFront에서 오리진 서버의 HTTP 3xx 상태 코드를 처리하는 방법 (p. 250)
- CloudFront에서 오리진의 HTTP 4xx 및 5xx 상태 코드를 처리 및 캐싱하는 방법 (p. 250)

Amazon S3 오리진에 대한 요청 및 응답 동작

주제

- CloudFront에서 HTTP 및 HTTPS 요청을 처리하는 방법 (p. 230)
- CloudFront에서 요청을 처리하고 Amazon S3 오리진 서버에 요청을 전달하는 방법 (p. 230)
- CloudFront에서 Amazon S3 오리진 서버의 요청을 처리하는 방법 (p. 234)

CloudFront에서 HTTP 및 HTTPS 요청을 처리하는 방법

Amazon S3 오리진의 경우 CloudFront에서는 기본적으로 CloudFront 배포의 객체에 대한 HTTP 및 HTTPS 프로토콜 모두의 요청을 수락합니다. 그런 뒤 CloudFront에서는 요청이 전달된 프로토콜과 같은 프로토콜을 사용하여 이 요청을 Amazon S3 버킷에 전달합니다.

사용자 지정 오리진의 경우 배포를 만들 때 HTTP 전용 또는 최종 사용자가 사용한 프로토콜과 일치 중에 CloudFront에서 오리진에 액세스하는 방법을 지정할 수 있습니다. CloudFront에서 사용자 지정 오리진에 대한 HTTP 및 HTTPS 요청을 처리하는 방법에 대한 자세한 내용은 프로토콜 (p. 243) 단원을 참조하십시오.

최종 사용자가 HTTPS만 사용하여 객체에 액세스할 수 있도록 웹 배포를 제한하는 방법에 대한 자세한 내용은 CloudFront에서 HTTPS 사용 (p. 92) 단원을 참조하십시오. (이 옵션은 RTMP 프로토콜을 사용하는 RTMP 배포에는 적용되지 않습니다.)

> **Note**
>
> HTTPS 요청에 대한 요금은 HTTP 요청에 대한 요금에 비해 높습니다. 부과되는 요율에 대한 자세한 내용은 CloudFront 요금제 페이지를 참조하십시오.

CloudFront에서 요청을 처리하고 Amazon S3 오리진 서버에 요청을 전달하는 방법

이 주제에는 CloudFront 에서 최종 사용자 요청을 처리하고 이 요청을 Amazon S3 오리진에 전달하는 방법에 대한 자세한 내용이 포함되어 있습니다.

주제

- 캐싱 시간 및 최소 TTL (p. 231)
- 클라이언트 IP 주소 (p. 231)
- 조건부 GET (p. 231)
- 쿠키 (p. 232)
- CORS(Cross-Origin Resource Sharing) (p. 232)
- 본문이 포함되는 GET 요청 (p. 232)
- HTTP 메소드 (p. 232)
- CloudFront에서 제거하거나 업데이트하는 HTTP 요청 헤더 (p. 233)
- 요청의 최대 길이 및 최대 URL 길이 (p. 233)
- OCSP 스테이플링 (p. 233)
- 프로토콜 (p. 233)
- 쿼리 문자열 (p. 233)
- 오리진 응답 제한 시간 (p. 234)
- 동일 객체에 대한 동시 요청(트래픽 스파이크) (p. 234)

캐싱 시간 및 최소 TTL

웹 배포의 경우 CloudFront에서 다른 요청을 오리진에 전달하기 전에 객체를 CloudFront 캐시에 보관하는 시간을 제어하려면

- `Cache-Control` 또는 `Expires` 헤더 파일을 각 객체에 추가하도록 오리진을 구성합니다.
- CloudFront 캐시 동작에 최소 TTL 값을 지정합니다.
- 기본값인 24시간을 사용합니다.

자세한 내용은 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204) 단원을 참조하십시오.

클라이언트 IP 주소

최종 사용자가 CloudFront에 요청을 보내고 `X-Forwarded-For` 요청 헤더를 포함하지 않는 경우, CloudFront는 TCP 연결에서 최종 사용자의 IP 주소를 가져오고 IP 주소를 포함하는 `X-Forwarded-For` 헤더를 추가하고 오리진에 요청을 전달합니다. 예를 들어 CloudFront가 IP TCP 연결에서 주소 `192.0.2.2`를 가져오면 오리진에 다음 헤더를 전달합니다.

```
X-Forwarded-For: 192.0.2.2
```

최종 사용자가 CloudFront에 요청을 보내고 `X-Forwarded-For` 요청 헤더를 포함하는 경우, CloudFront는 TCP 연결에서 최종 사용자의 IP 주소를 가져와 `X-Forwarded-For` 헤더의 끝에 첨부하고 오리진에 요청을 전달합니다. 예를 들어 최종 사용자 요청에 `X-Forwarded-For: 192.0.2.4,192.0.2.3`가 포함되고 CloudFront가 TCP 연결에서 IP 주소 `192.0.2.2`를 가져오면 오리진에 다음 헤더를 전달합니다.

```
X-Forwarded-For: 192.0.2.4,192.0.2.3,192.0.2.2
```

> **Note**
>
> `X-Forwarded-For` 헤더에는 IPv4 주소(예: 192.0.2.44)와 IPv6 주소(예: 2001:0db8:85a3:0000:0000:8a2e:0370:7334)가 포함됩니다.

조건부 GET

CloudFront는 엣지 캐시에서 만료된 객체에 대한 요청을 받으면 이 요청을 Amazon S3 오리진으로 전달하여 최신 버전의 객체를 가져오거나 CloudFront 엣지 캐시에 이미 최신 버전이 있는지 Amazon S3의 확인을 받습니다. Amazon S3에서는 처음에 CloudFront에 객체를 보낼 때 `ETag` 값과 `LastModified` 값을 응답에

포함하여 보냈습니다. CloudFront에서 Amazon S3에 전달하는 새 요청에서 CloudFront는 다음 중 하나 또는 둘 모두를 추가합니다.

- 만료된 버전의 객체에 대한 `If-Match` 값을 포함하는 `If-None-Match` 또는 `ETag` 헤더
- 만료된 버전의 객체에 대한 `If-Modified-Since` 값을 포함하는 `LastModified` 헤더

Amazon S3에서는 이 정보를 사용하여 객체가 업데이트되는지 여부와, 그에 따라 전체 객체를 CloudFront에 반환할지 아니면 HTTP 304 상태 코드만 반환할지(수정되지 않음) 여부를 결정합니다.

쿠키

Amazon S3에서는 쿠키를 처리하지 않습니다. 쿠키를 Amazon S3 오리진에 전달하도록 캐시 동작을 구성하는 경우, CloudFront에서는 이 쿠키를 전달하지만 Amazon S3에서는 이를 무시합니다. 향후 동일한 객체에 대한 모든 요청은 쿠키 변경 여부와 관계없이 캐시에 들어 있는 기존 객체로 처리합니다.

CORS(Cross-Origin Resource Sharing)

CloudFront에서 Amazon S3 cross-origin 리소스 공유 설정을 지키도록 하려는 경우, 선택한 헤더를 Amazon S3에 전달하도록 CloudFront를 구성합니다. 자세한 내용은 요청 헤더에 따라 콘텐츠 캐싱 (p. 200) 단원을 참조하십시오.

본문이 포함되는 GET 요청

최종 사용자 `GET` 요청에 본문이 포함되는 경우, CloudFront에서는 HTTP 상태 코드 403(금지됨)을 최종 사용자에게 반환합니다.

HTTP 메소드

지원되는 모든 HTTP 메서드를 처리하도록 CloudFront를 구성하는 경우, CloudFront에서는 최종 사용자의 다음 요청을 수락하고 이를 Amazon S3 오리진에 전달합니다.

- `DELETE`
- `GET`
- `HEAD`
- `OPTIONS`
- `PATCH`
- `POST`
- `PUT`

CloudFront에서는 `GET` 및 `HEAD` 요청에 대한 응답을 항상 캐싱합니다. 또한 `OPTIONS` 요청에 대한 응답을 캐싱하도록 CloudFront를 구성할 수도 있습니다. CloudFront에서는 다른 메서드를 사용하는 요청에 대한 응답을 캐싱하지 않습니다.

Amazon S3 버킷을 배포용 오리진으로 사용하는 경우와 CloudFront 원본 액세스 ID를 사용하는 경우, `POST` 요청은 일부 Amazon S3 리전에서 지원되지 않으며 해당 리전의 `PUT` 요청에는 추가 헤더가 필요합니다. 자세한 내용은 서명 버전 4 인증만 지원하는 Amazon S3 리전에서 OAI 사용 (p. 182) 단원을 참조하십시오.

다중 파트 업로드를 사용하여 객체를 Amazon S3 버킷에 추가하려는 경우, CloudFront 원본 액세스 ID를 배포에 추가하고 이 원본 액세스 ID에 필요한 사용 권한을 부여해야 합니다. 자세한 내용은 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오.

> **Important**
>
> CloudFront에서 지원하는 모든 HTTP 메소드를 Amazon S3에서 허용하고 전달하도록 CloudFront를 구성하는 경우, Amazon S3 콘텐츠에 대한 액세스를 제한하는 CloudFront 원본 액세스 ID를 만

들어 이 원본 액세스 ID에 필수 사용 권한을 부여해야 합니다. 예를 들어, `PUT`을 사용할 의도로 이러한 메서드를 허용 및 전달하도록 CloudFront를 구성한 경우, 최종 사용자에 의해 삭제되는 것을 원치 않는 리소스를 이들이 삭제할 수 없도록 `DELETE` 요청을 적절히 처리하여 Amazon S3 버킷 정책 또는 ACL을 구성해야 합니다. 자세한 내용은 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오.

Amazon S3에서 지원되는 동작에 대한 자세한 내용은 Amazon S3 설명서를 참조하십시오.

CloudFront에서 제거하거나 업데이트하는 HTTP 요청 헤더

CloudFront는 Amazon S3 오리진에 요청을 전달하기 전에 몇몇 헤더를 제거하거나 업데이트합니다. 대부분의 헤더에서 이 동작은 사용자 지정 오리진에서의 동작과 같습니다. HTTP 요청 헤더의 전체 목록과 CloudFront에서 이를 처리하는 방법은 HTTP 요청 헤더 및 CloudFront 동작(사용자 지정 및 S3 오리진) (p. 239) 단원을 참조하십시오.

요청의 최대 길이 및 최대 URL 길이

경로, 쿼리 문자열(있는 경우), 헤더를 모두 포함한 요청의 최대 길이는 20,480바이트입니다.

CloudFront에서는 이 요청으로부터 URL을 구성합니다. 이 URL의 최대 길이는 8,192바이트입니다.

요청 또는 URL이 이 최대값을 초과할 경우 CloudFront에서는 HTTP 요청 코드 413(요청 개체가 너무 큼)을 반환한 후 최종 사용자와의TCP 연결을 종료합니다.

OCSP 스테이플링

최종 사용자가 객체에 대한 HTTPS 요청을 제출할 때 CloudFront 또는 최종 사용자는 CA(인증 기관)을 통해 도메인의 SSL 인증서가 해지되지 않았는지 확인해야 합니다. OCSP 스테이플링은 CloudFront에서 인증서의 유효성을 검사하고 CA로부터 응답을 캐싱할 수 있도록 함으로써 클라이언트가 CA를 통해 직접 인증서의 유효성을 검사하지 않아도 되므로 인증서 유효성 검사 속도가 향상됩니다.

OSCP 스테이플링의 성능 개선은 CloudFront에서 같은 도메인 내의 객체에 대해 많은 HTTPS 요청을 받은 경우 더욱 확연히 드러납니다. CloudFront 엣지 로케이션의 각 서버는 개별적인 유효성 검사 요청을 제출해야 합니다. CloudFront에서 같은 도메인에 대해 다수의 HTTPS 요청을 받은 경우, 엣지 로케이션의 각 서버에서는 곧 CA로부터 SSL 핸드셰이크의 패킷에 "스테이플"할 수 있다는 응답을 받습니다. 최종 사용자가 인증서가 유효하다는 조건을 충족하면 CloudFront에서는 요청된 객체를 제공할 수 있습니다. 배포가 CloudFront 엣지 로케이션에서 많은 양의 트래픽을 받지 않는 경우, 새로운 요청은 아직 CA를 통해 인증서의 유효성을 검사하지 않은 서버로 리디렉션될 가능성이 높습니다. 그러한 경우, 최종 사용자는 유효성 검사 단계를 개별적으로 수행하고 CloudFront 서버에서는 객체를 제공합니다. 해당 CloudFront 서버에서는 또한 유효성 검사 요청을 CA에 제출하고, 다음에 동일한 도메인 이름을 포함한 요청을 수신하면 CA로부터 유효성 검사 응답을 받습니다.

프로토콜

CloudFront에서는 HTTP 또는 HTTPS 요청을 최종 사용자 요청의 프로토콜(HTTP 또는 HTTPS)을 바탕으로 오리진 서버에 전달합니다.

> **Important**
>
> 웹 사이트 엔드포인트로 Amazon S3 버킷이 구성되어 있는 경우, Amazon S3에서 해당 구성으로 HTTPS 연결을 지원하지 않으므로 HTTPS를 사용하여 오리진과 통신하도록 CloudFront를 구성할 수 없습니다.

쿼리 문자열

웹 배포의 경우 CloudFront에서 쿼리 문자열 파라미터를 Amazon S3 오리진에 전달할지 여부를 구성할 수 있습니다. RTMP 배포의 경우 CloudFront에서는 쿼리 문자열 파라미터를 전달하지 않습니다. 자세한 내용은 쿼리 문자열 파라미터 기반의 콘텐츠 캐싱 (p. 196) 단원을 참조하십시오.

오리진 응답 제한 시간

오리진 응답 제한 시간(오리진 읽기 제한 시간 또는 오리진 요청 제한 시간이라고도 함)은 다음 두 값에 모두 적용됩니다.

- CloudFront가 Amazon S3에 요청을 전달한 후 응답을 기다리는 시간(초)
- CloudFront가 S3로부터 응답 패킷을 수신한 후 다음 패킷을 수신할 때까지 대기하는 시간(초)

CloudFront 동작은 HTTP 메서드에 따라 달라집니다.

- `GET` 및 `HEAD` 요청 – Amazon S3에서 30초 내에 응답하지 않거나 30초 동안 응답이 중지된 경우, CloudFront에서는 연결을 끊고 오리진에 두 번 더 연결을 시도합니다. 오리진이 세 번째 시도에 응답하지 않는 경우, CloudFront에서는 동일한 CloudFront 오리진의 콘텐츠에 대해 다른 요청을 받을 때까지 다시 시도하지 않습니다.
- `DELETE`, `OPTIONS`, `PATCH`, `PUT`, `POST` 요청 – Amazon S3에서 30초 내에 응답하지 않는 경우, CloudFront에서는 연결을 끊고 오리진에 다시 연결을 시도하지 않습니다. 필요한 경우 클라이언트는 요청을 다시 제출할 수 있습니다.

모든 요청에서 CloudFront는 S3와 연결 설정을 시도합니다. 10초 이내에 연결이 되지 않을 경우 CloudFront는 연결을 끊고 S3에 두 번 더 연결을 시도합니다. 오리진이 세 번째 시도에 응답하지 않는 경우, CloudFront에서는 동일한 오리진의 콘텐츠에 대해 다른 요청을 받을 때까지 다시 시도하지 않습니다.

S3에 대한 응답 제한 시간은 변경할 수 없습니다.

동일 객체에 대한 동시 요청(트래픽 스파이크)

CloudFront 엣지 로케이션에서 객체에 대한 요청을 받을 때 객체가 현재 캐시에 있지 않거나 객체가 만료된 경우, CloudFront에서는 즉시 요청을 Amazon S3 오리진으로 보냅니다. 트래픽 스파이크가 있는 경우, 즉 Amazon S3에서 첫 번째 요청에 응답하기 전에 동일 객체에 대한 추가 요청이 엣지 로케이션에 도착하는 경우, CloudFront는 객체에 대한 추가 요청을 오리진에 전달하기 전에 작업을 일시 중단합니다. 일반적으로는 후속 요청에 응답하기 전에 첫 번째 요청에 대한 응답이 CloudFront 엣지 로케이션에 도착하게 됩니다. 이러한 일시 중단은 Amazon S3에 불필요하게 로드가 걸리는 것을 줄여 줍니다. 요청 헤더나 쿼리 문자열에 따라 캐싱하도록 CloudFront를 구성하는 등의 이유로 추가 요청이 동일하지 않은 경우, CloudFront에서는 모든 고유한 요청을 오리진에 전달합니다.

오리진의 응답에 `Cache-Control: no-cache` 헤더가 포함된 경우, CloudFront에서는 일반적으로 동일 객체에 대한 다음 요청을 이 오리진에 전달하여 객체가 업데이트되었는지 여부를 확인합니다. 그러나 트래픽 스파이크가 발생하여 CloudFront에서 첫 번째 요청을 오리진에 전달한 후에 작업이 일시 중단된 경우, CloudFront에서 오리진의 응답을 받기 전에 복수 개의 최종 사용자 요청이 도착할 수 있습니다. CloudFront에서 `Cache-Control: no-cache` 헤더가 포함된 응답을 받은 경우, 원래 요청한 최종 사용자와 일시 중지 중에 객체를 요청한 모든 최종 사용자에 대한 응답으로 객체를 전달합니다. 오리진에서 응답이 도착한 후에는 CloudFront에서 동일 객체에 대한 다음 최종 사용자 요청을 오리진에 전달합니다. CloudFront 액세스 로그에서, 첫 번째 요청은 `x-edge-result-type` 열에서 `Miss`로 식별되고 CloudFront에서 일시 중지 중에 받은 모든 후속 요청은 `Hit`로 식별됩니다. 로그 파일 형식에 대한 자세한 내용은 웹 배포 로그 파일 형식 (p. 399) 단원을 참조하십시오.

CloudFront에서 Amazon S3 오리진 서버의 요청을 처리하는 방법

이 주제에는 CloudFront가 Amazon S3 오리진에서 요청을 처리하는 방법에 대한 자세한 내용이 포함되어 있습니다.

주제

- 취소된 요청 (p. 235)

- CloudFront에서 제거하거나 업데이트하는 HTTP 응답 헤더 (p. 235)
- 최대 파일 크기 (p. 235)
- 리디렉션 (p. 235)

취소된 요청

객체가 엣지 캐시에 있지 않은 경우 최종 사용자가 CloudFront가 오리진에서 객체를 가져온 뒤 요청된 객체를 제공하기 전에 브라우저 닫기 등으로 세션을 종료하면, CloudFront에서는 엣지 로케이션의 객체를 캐싱하지 않습니다.

CloudFront에서 제거하거나 업데이트하는 HTTP 응답 헤더

CloudFront는 Amazon S3 오리진에서 최종 사용자에게 응답을 전달하기 전에 다음 헤더 필드를 제거하거나 업데이트합니다.

- `Set-Cookie` – 쿠키를 전달하도록 CloudFront를 구성하는 경우, `Set-Cookie` 헤더 필드가 클라이언트에 전달됩니다. 자세한 내용은 쿠키를 기반으로 콘텐츠 캐싱 (p. 198) 단원을 참조하십시오.
- `Trailer`
- `Transfer-Encoding` – Amazon S3 오리진에서 이 헤더 필드를 반환하는 경우, CloudFront에서는 최종 사용자에게 응답을 반환하기 전에 `chunked`에 값을 설정합니다.
- `Upgrade`
- `Via` – CloudFront는 최종 사용자에 대한 응답으로 값을 다음과 같이 설정합니다.

  ```
  Via: http-version alphanumeric-string.cloudfront.net (CloudFront)
  ```

  예를 들어, 클라이언트가 HTTP/1.1을 통해 요청한 경우 값은 다음과 같습니다.

  ```
  Via: 1.1 1026589cc7887e7a0dc7827b4example.cloudfront.net (CloudFront)
  ```

최대 파일 크기

CloudFront에서 최종 사용자에게 반환하는 응답 본문의 최대 크기는 20GB입니다. 여기에는 `Content-Length` 헤더 값으로 지정하지 않은 조각난 전송 응답이 포함됩니다.

리디렉션

모든 요청을 다른 호스트 이름으로 리디렉션하도록 Amazon S3 버킷을 구성할 수 있습니다. 이 대상은 다른 Amazon S3 버킷 또는 HTTP 서버가 될 수 있습니다. 모든 요청을 리디렉션하도록 버킷을 구성하는 경우와 버킷이 CloudFront 배포에 대한 오리진인 경우, 배포의 도메인 이름(예: d111111abcdef8.cloudfront.net) 또는 배포와 연결된 대체 도메인 이름(CNAME)(예: example.com)을 사용하여 모든 요청을 CloudFront 배포에 리디렉션하도록 버킷을 구성하는 것이 좋습니다. 그렇지 않은 경우 최종 사용자는 CloudFront를 우회하도록 요청하고 객체는 새 오리진에서 직접 제공됩니다.

> **Note**
>
> 요청을 대체 도메인 이름으로 리디렉션하는 경우, CNAME 레코드를 추가하여 도메인에 대한 DNS 서비스 역시 업데이트해야 합니다. 자세한 내용은 대체 도메인 이름(CNAME)을 추가하여 파일에 대해 사용자 지정 URL 사용 (p. 66) 단원을 참조하십시오.

모든 요청을 리디렉션하도록 버킷을 구성했을 경우 다음과 같은 상황이 발생합니다.

1. 브라우저 등의 최종 사용자가 CloudFront에서 객체를 요청합니다.
2. CloudFront에서는 이 요청을 배포에 대한 오리진인 Amazon S3 버킷에 전달합니다.
3. Amazon S3에서는 HTTP 상태 코드 301(영구적으로 옮겨짐)과 함께 새 위치를 반환합니다.

4. CloudFront에서는 리디렉션 상태 코드와 새 위치를 캐싱하고 최종 사용자에게 값을 반환합니다. CloudFront는 이 리디렉션을 따라가지 않고 새 위치에서 객체를 가져옵니다.
5. 최종 사용자는 객체에 대한 다른 요청을 보내지만, 이때 최종 사용자는 CloudFront에서 가져온 새 위치를 지정합니다.
   - Amazon S3 버킷에서 모든 요청을 CloudFront 배포로 리디렉션하는 경우, CloudFront에서는 배포의 도메인 이름 또는 대체 도메인 이름을 사용하여 새 위치의 Amazon S3 버킷 또는 HTTP 서버에서 객체를 요청합니다. 새 위치에서 객체를 반환하는 경우, CloudFront에서는 최종 사용자에게 이를 반환하고 엣지 로케이션에 이를 캐싱합니다.
   - Amazon S3 버킷에서 요청을 다른 위치로 리디렉션하는 경우, 두 번째 요청은 CloudFront를 우회합니다. 새 위치의 Amazon S3 버킷 또는 HTTP 서버에서 최종 사용자에게 직접 객체를 반환하므로, 객체는 CloudFront 엣지 캐시에서 캐싱되지 않습니다.

사용자 지정 오리진에 대한 요청 및 응답 동작

주제

- CloudFront에서 요청을 처리하고 사용자 지정 오리진 서버에 요청을 전달하는 방법 (p. 236)
- CloudFront에서 사용자 지정 오리진 서버의 요청을 처리하는 방법 (p. 244)

CloudFront에서 요청을 처리하고 사용자 지정 오리진 서버에 요청을 전달하는 방법

이 주제에는 CloudFront에서 최종 사용자 요청을 처리하고 이 요청을 사용자 지정 오리진에 전달하는 방법에 대한 자세한 내용이 포함되어 있습니다.

주제

- 인증 (p. 237)
- 캐싱 시간 및 최소 TTL (p. 237)
- 클라이언트 IP 주소 (p. 237)
- 클라이언트측 SSL 인증 (p. 238)
- 압축 (p. 238)
- 조건부 요청 (p. 238)
- 쿠키 (p. 238)
- CORS(Cross-Origin Resource Sharing) (p. 238)
- 암호화 (p. 238)
- 본문이 포함되는 GET 요청 (p. 239)
- HTTP 메소드 (p. 239)
- HTTP 요청 헤더 및 CloudFront 동작(사용자 지정 및 S3 오리진) (p. 239)
- HTTP 버전 (p. 242)
- 요청의 최대 길이 및 최대 URL 길이 (p. 242)
- OCSP 스테이플링 (p. 242)
- 지속적인 연결 (p. 243)
- 프로토콜 (p. 243)
- 쿼리 문자열 (p. 243)
- 오리진 응답 제한 시간 (p. 243)
- 동일 객체에 대한 동시 요청(트래픽 스파이크) (p. 244)

- User-Agent 헤더 (p. 244)

인증

`DELETE`, `GET`, `HEAD`, `PATCH`, `POST`, `PUT` 요청에 대해 `Authorization` 헤더를 오리진에 전달하도록 CloudFront를 구성하는 경우, 클라이언트 인증을 요청하도록 오리진 서버를 구성할 수 있습니다.

`OPTIONS` 요청에 대해서는 다음 CloudFront 설정을 사용하는 경우에만 클라이언트 인증을 요청하도록 오리진 서버를 구성할 수 있습니다.

- `Authorization` 헤더를 오리진에 전달하도록 CloudFront를 구성합니다.
- `OPTIONS` 요청에 대한 응답을 캐싱하지 않도록 CloudFront를 구성합니다.

HTTP 또는 HTTPS를 사용하여 오리진에 요청을 전달하도록 CloudFront를 구성할 수 있습니다. 자세한 내용은 CloudFront에서 HTTPS 사용 (p. 92) 단원을 참조하십시오.

캐싱 시간 및 최소 TTL

웹 배포의 경우 CloudFront에서 다른 요청을 오리진에 전달하기 전에 객체를 CloudFront 캐시에 보관하는 시간을 제어하려면

- `Cache-Control` 또는 `Expires` 헤더 파일을 각 객체에 추가하도록 오리진을 구성합니다.
- CloudFront 캐시 동작에 최소 TTL 값을 지정합니다.
- 기본값인 24시간을 사용합니다.

자세한 내용은 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204) 단원을 참조하십시오.

클라이언트 IP 주소

최종 사용자가 CloudFront에 요청을 보내고 `X-Forwarded-For` 요청 헤더를 포함하지 않는 경우, CloudFront는 TCP 연결에서 최종 사용자의 IP 주소를 가져오고 IP 주소를 포함하는 `X-Forwarded-For` 헤더를 추가하고 오리진에 요청을 전달합니다. 예를 들어 CloudFront가 IP TCP 연결에서 주소 `192.0.2.2`를 가져오면 오리진에 다음 헤더를 전달합니다.

```
X-Forwarded-For: 192.0.2.2
```

최종 사용자가 CloudFront에 요청을 보내고 `X-Forwarded-For` 요청 헤더를 포함하는 경우, CloudFront는 TCP 연결에서 최종 사용자의 IP 주소를 가져와 `X-Forwarded-For` 헤더의 끝에 첨부하고 오리진에 요청을 전달합니다. 예를 들어 최종 사용자 요청에 `X-Forwarded-For: 192.0.2.4,192.0.2.3`가 포함되고 CloudFront가 TCP 연결에서 IP 주소 `192.0.2.2`를 가져오면 오리진에 다음 헤더를 전달합니다.

```
X-Forwarded-For: 192.0.2.4,192.0.2.3,192.0.2.2
```

로드 밸런서(Elastic Load Balancing 포함), 웹 애플리케이션 방화벽, 역방향 프록시, 침입 방지 시스템 및 API 게이트웨이와 같은 일부 애플리케이션들은 해당 요청을 전달한 CloudFront 엣지 서버의 IP 주소를 `X-Forwarded-For` 헤더의 끝에 추가합니다. 예를 들어 CloudFront가 ELB에 전달하는 요청에 `X-Forwarded-For: 192.0.2.2`를 포함시키고 CloudFront 엣지 서버의 IP 주소가 192.0.2.199인 경우, EC2 인스턴스가 수신하는 요청에는 다음과 같은 헤더가 들어 있습니다.

```
X-Forwarded-For: 192.0.2.2,192.0.2.199
```

> **Note**
>
> `X-Forwarded-For` 헤더에는 IPv4 주소(예: 192.0.2.44)와 IPv6 주소(예: 2001:0db8:85a3:0000:0000:8a2e:0370:7334)가 포함됩니다.

클라이언트측 SSL 인증

CloudFront는 클라이언트측 SSL 인증서와 클라이언트 인증을 지원하지 않습니다. 오리진이 클라이언트측 인증서를 요청하는 경우 CloudFront에서는 이 요청을 삭제합니다.

압축

CloudFront에서는 `Accept-Encoding` 필드 값이 `"identity"` 및 `"gzip"`인 요청을 전달합니다. 자세한 내용은 압축된 파일 제공 (p. 87) 단원을 참조하십시오.

조건부 요청

CloudFront는 엣지 캐시에서 만료된 객체에 대한 요청을 받으면 이 요청을 오리진으로 전달하여 최신 버전의 객체를 가져오거나 CloudFront 엣지 캐시에 이미 최신 버전이 있는지 오리진의 확인을 받습니다. 대개 오리진에서 마지막에 CloudFront에 객체를 보낼 때는 `ETag` 값, `LastModified` 값 또는 두 값 모두를 응답에 포함하여 보냅니다. CloudFront에서 오리진에 전달하는 새 요청에서 CloudFront는 다음 중 하나 또는 둘 모두를 추가합니다.

- 만료된 버전의 객체에 대한 `If-Match` 값을 포함하는 `If-None-Match` 또는 `ETag` 헤더
- 만료된 버전의 객체에 대한 `If-Modified-Since` 값을 포함하는 `LastModified` 헤더

오리진에서는 이 정보를 사용하여 객체가 업데이트되는지 여부와, 그에 따라 전체 객체를 CloudFront에 반환할지 아니면 HTTP 304 상태 코드(수정되지 않음)만 반환할지 여부를 결정합니다.

쿠키

쿠키를 오리진에 전달하도록 CloudFront를 구성할 수 있습니다. 자세한 내용은 쿠키를 기반으로 콘텐츠 캐싱 (p. 198) 단원을 참조하십시오.

CORS(Cross-Origin Resource Sharing)

CloudFront에서 cross-origin 리소스 공유 설정을 지키도록 하려는 경우, `Origin` 헤더를 오리진에 전달하도록 CloudFront를 구성합니다. 자세한 내용은 요청 헤더에 따라 콘텐츠 캐싱 (p. 200) 단원을 참조하십시오.

암호화

최종 사용자는 HTTPS를 사용하여 CloudFront에 요청을 전송하도록 하고 CloudFront에는 최종 사용자에 의해 사용된 프로토콜을 사용하여 사용자 지정 오리진에 요청을 전달하도록 할 수 있습니다. 자세한 내용은 다음 배포 설정을 참조하십시오.

- Viewer Protocol Policy (p. 45)
- 오리진 프로토콜 정책 (p. 41)

CloudFront에서는 SSLv3, TLSv1.0, TLSv1.1 및 TLSv1.2 프로토콜을 사용하여 HTTPS 요청을 오리진 서버에 전달합니다. 사용자 지정 오리진의 경우, 오리진과 통신할 때 CloudFront에서 사용하려는 SSL 프로토콜을 선택할 수 있습니다.

- CloudFront 콘솔을 사용하는 경우, 오리진 SSL 프로토콜 확인란을 사용하여 프로토콜을 선택합니다. 자세한 내용은 배포 만들기 (p. 35) 단원을 참조하십시오.
- CloudFront API를 사용하는 경우, `OriginSslProtocols` 요소를 사용하여 프로토콜을 지정하십시오. 자세한 내용은 Amazon CloudFront API Reference의 OriginSslProtocols 및 DistributionConfig 단원을 참조하십시오.

오리진이 Amazon S3 버킷인 경우, CloudFront는 항상 TLSv1.2를 사용합니다.

> **Important**
>
> 다른 버전의 SSL 및 TLS는 지원되지 않습니다.

CloudFront에서 HTTPS를 사용하는 방법은 CloudFront에서 HTTPS 사용 (p. 92) 단원을 참조하십시오. 최종 사용자와 CloudFront 간의 HTTPS 통신 및 CloudFront와 오리진 간의 HTTPS 통신에 대해 CloudFront가 지원하는 암호 목록은 최종 사용자와 CloudFront 간 통신에 지원되는 SSL/TLS 프로토콜 및 암호 (p. 99) 단원을 참조하십시오.

본문이 포함되는 GET 요청

최종 사용자 `GET` 요청에 본문이 포함되는 경우, CloudFront에서는 HTTP 상태 코드 403(금지됨)을 최종 사용자에게 반환합니다.

HTTP 메소드

지원되는 모든 HTTP 메서드를 처리하도록 CloudFront를 구성하는 경우, CloudFront에서는 최종 사용자의 다음 요청을 수락하고 이를 사용자 지정 오리진에 전달합니다.

- `DELETE`
- `GET`
- `HEAD`
- `OPTIONS`
- `PATCH`
- `POST`
- `PUT`

CloudFront에서는 `GET` 및 `HEAD` 요청에 대한 응답을 항상 캐싱합니다. 또한 `OPTIONS` 요청에 대한 응답을 캐싱하도록 CloudFront를 구성할 수도 있습니다. CloudFront에서는 다른 메서드를 사용하는 요청에 대한 응답을 캐싱하지 않습니다.

사용자 지정 오리진에서 이러한 메소드를 처리할지 여부를 구성하는 방법에 대한 자세한 내용은 오리진에 대한 설명서를 참조하십시오.

> **Important**
>
> CloudFront에서 지원되는 모든 HTTP 메서드를 허용하고 오리진에 전달하도록 CloudFront를 구성한 경우, 모든 메서드를 처리하도록 오리진 서버를 구성합니다. 예를 들어, `POST`를 사용할 의도로 이러한 메서드를 허용 및 전달하도록 CloudFront를 구성한 경우, 최종 사용자에 의해 삭제되는 것을 원치 않는 리소스를 이들이 삭제할 수 없도록 `DELETE` 요청을 적절히 처리하여 오리진 서버를 구성해야 합니다. 자세한 내용은 HTTP 서버에 대한 설명서를 참조하십시오.

HTTP 요청 헤더 및 CloudFront 동작(사용자 지정 및 S3 오리진)

다음 테이블에는 사용자 지정 오리진과 Amazon S3 오리진 모두에 전달할 수 있는 HTTP 요청 헤더가 나열되어 있습니다(예외 참조). 각 헤더의 테이블에는 다음에 대한 정보가 포함되어 있습니다.

- 헤더를 오리진에 전달하도록 CloudFront를 구성하지 않은 경우의 CloudFront 동작으로, CloudFront에서 헤더 값에 따라 객체를 캐싱하는 원인이 됩니다.
- 해당 헤더에 대해 헤더 값에 따라 객체를 캐싱하도록 CloudFront를 구성할 수 있는지 여부.

  `Date` 및 `User-Agent` 헤더의 값에 따라 객체를 캐싱하도록 CloudFront를 구성할 수 있지만, 권장되지는 않습니다. 이러한 헤더는 여러 가지 값을 가질 수 있으며, 헤더의 값에 따른 캐싱으로 인해 CloudFront에서 오리진에 전달되는 요청이 눈에 띄게 증가할 수 있습니다.

헤더 값에 따른 캐싱에 대한 자세한 내용은 요청 헤더에 따라 콘텐츠 캐싱 (p. 200) 단원을 참조하십시오.

헤더	헤더 값에 따라 캐싱하도록 CloudFront를 구성하지 않은 경우의 동작	헤더 값에 따른 캐싱이 지원되는지 여부
기타 정의 헤더	CloudFront에서 오리진에 헤더를 전달합니다.	예
`Accept`	CloudFront에서 헤더를 제거합니다.	예
`Accept-Charset`	CloudFront에서 헤더를 제거합니다.	예
`Accept-Encoding`	값에 `gzip`이 포함되어 있는 경우, CloudFront에서 `Accept-Encoding: gzip`을 오리진에 전달합니다. 이 값에 `gzip`이 포함되어 있지 않은 경우, CloudFront에서는 오리진으로 요청을 전달하기 전에 `Accept-Encoding` 헤더 필드를 제거합니다.	예
`Accept-Language`	CloudFront에서 헤더를 제거합니다.	예
`Authorization`	• `GET` 및 `HEAD` 요청 – CloudFront에서는 요청을 오리진에 전달하기 전에 `Authorization` 헤더 필드를 제거합니다. • `OPTIONS` 요청 – `OPTIONS` 요청에 대한 응답을 캐싱하도록 CloudFront를 구성한 경우, CloudFront에서는 요청을 오리진에 전달하기 전에 `Authorization` 헤더 필드를 제거합니다. OPTIONS 요청에 대한 응답을 캐싱하도록 CloudFront를 구성하지 않은 경우, CloudFront에서는 `Authorization` 헤더 필드를 오리진에 전달합니다. • `DELETE`, `PATCH`, `POST`, `PUT` 요청 – CloudFront에서는 요청을 오리진에 전달하기 전에 헤더 필드를 제거하지 않습니다.	예
`Cache-Control`	CloudFront에서 오리진에 헤더를 전달합니다.	아니요
`CloudFront-Forwarded-Proto`	요청을 오리진에 전달하기 전에는 CloudFront에서 헤더를 추가하지 않습니다. 자세한 내용은 요청의 프로토콜을 기반으로 캐싱하도록 구성 (p. 203) 단원을 참조하십시오.	예
`CloudFront-Is-Desktop-Viewer`	요청을 오리진에 전달하기 전에는 CloudFront에서 헤더를 추가하지 않습니다. 자세한 내용은 디바이스 유형을 기반으로 캐싱하도록 구성 (p. 203) 단원을 참조하십시오.	예
`CloudFront-Is-Mobile-Viewer`	요청을 오리진에 전달하기 전에는 CloudFront에서 헤더를 추가하지 않습니다. 자세한 내용은 디바이스 유형을 기반으로 캐싱하도록 구성 (p. 203) 단원을 참조하십시오.	예
`CloudFront-Is-Tablet-Viewer`	요청을 오리진에 전달하기 전에는 CloudFront에서 헤더를 추가하지 않습니다. 자세한 내용은 디바이스 유형을 기반으로 캐싱하도록 구성 (p. 203) 단원을 참조하십시오.	예

헤더	헤더 값에 따라 캐싱하도록 CloudFront를 구성하지 않은 경우의 동작	헤더 값에 따른 캐싱이 지원되는지 여부
`CloudFront-Viewer-Country`	요청을 오리진에 전달하기 전에는 CloudFront에서 헤더를 추가하지 않습니다.	예
`Connection`	CloudFront에서는 요청을 오리진에 전달하기 전에 `Connection: Keep-Alive`로 이 헤더를 대체합니다.	아니요
`Content-Length`	CloudFront에서 오리진에 헤더를 전달합니다.	아니요
`Content-MD5`	CloudFront에서 오리진에 헤더를 전달합니다.	예
`Content-Type`	CloudFront에서 오리진에 헤더를 전달합니다.	예
`Cookie`	쿠키를 전달하도록 CloudFront를 구성한 경우, `Cookie` 헤더 필드가 오리진에 전달됩니다. 그렇지 않은 경우 CloudFront에서는 `Cookie` 헤더 필드를 제거합니다. 자세한 내용은 쿠키를 기반으로 콘텐츠 캐싱 (p. 198) 단원을 참조하십시오.	아니요
`Date`	CloudFront에서 오리진에 헤더를 전달합니다.	지원되나 권장되지 않음
`Expect`	CloudFront에서 헤더를 제거합니다.	예
`From`	CloudFront에서 오리진에 헤더를 전달합니다.	예
`Host`	CloudFront에서 값을 요청된 객체와 연결된 오리진의 도메인 이름에 설정합니다. Amazon S3 또는 MediaStore 오리진에 대한 호스트 헤더를 기반으로 캐시할 수 없습니다.	예(사용자 지정) 아니요(S3 및 MediaStore)
`If-Match`	CloudFront에서 오리진에 헤더를 전달합니다.	예
`If-Modified-Since`	CloudFront에서 오리진에 헤더를 전달합니다.	예
`If-None-Match`	CloudFront에서 오리진에 헤더를 전달합니다.	예
`If-Range`	CloudFront에서 오리진에 헤더를 전달합니다.	예
`If-Unmodified-Since`	CloudFront에서 오리진에 헤더를 전달합니다.	예
`Max-Forwards`	CloudFront에서 오리진에 헤더를 전달합니다.	아니요
`Origin`	CloudFront에서 오리진에 헤더를 전달합니다.	예
`Pragma`	CloudFront에서 오리진에 헤더를 전달합니다.	아니요
`Proxy-Authenticate`	CloudFront에서 헤더를 제거합니다.	아니요
`Proxy-Authorization`	CloudFront에서 헤더를 제거합니다.	아니요
`Proxy-Connection`	CloudFront에서 헤더를 제거합니다.	아니요
`Range`	CloudFront에서 오리진에 헤더를 전달합니다. 자세한 내용은 CloudFront에서 객체에 대한 부분적인 요청을 처리하는 방법(Range GET) (p. 214) 단원을 참조하십시오.	기본적으로 지원됨
`Referer`	CloudFront에서 헤더를 제거합니다.	예

헤더	헤더 값에 따라 캐싱하도록 CloudFront를 구성하지 않은 경우의 동작	헤더 값에 따른 캐싱이 지원되는지 여부
`Request-Range`	CloudFront에서 오리진에 헤더를 전달합니다.	아니요
`TE`	CloudFront에서 헤더를 제거합니다.	아니요
`Trailer`	CloudFront에서 헤더를 제거합니다.	아니요
`Transfer-Encoding`	CloudFront에서 오리진에 헤더를 전달합니다.	아니요
`Upgrade`	CloudFront에서는 WebSocket 연결이 설정되지 않았을 경우 헤더를 제거합니다.	아니요 (WebSocket 연결 제외)
`User-Agent`	CloudFront에서 이 헤더의 값을 `Amazon CloudFront`로 대체합니다. CloudFront에서 사용자가 이용 중인 디바이스에 따라 콘텐츠를 캐싱하도록 하려는 경우, 디바이스 유형을 기반으로 캐싱하도록 구성 (p. 203) 단원을 참조하십시오.	지원되나 권장되지 않음
`Via`	CloudFront에서 오리진에 헤더를 전달합니다.	예
`Warning`	CloudFront에서 오리진에 헤더를 전달합니다.	예
`X-Amz-Cf-Id`	CloudFront에서는 요청을 오리진에 전달하기 전에 최종 사용자 요청에 헤더를 추가합니다. 헤더 값에는 요청을 고유하게 식별하는 암호화된 문자열이 포함됩니다.	아니요
`X-Edge-*`	CloudFront는 모든 `X-Edge-*` 헤더를 삭제합니다.	아니요
`X-Forwarded-For`	CloudFront에서 오리진에 헤더를 전달합니다. 자세한 내용은 클라이언트 IP 주소 (p. 237) 단원을 참조하십시오.	예
`X-Forwarded-Proto`	CloudFront에서 헤더를 제거합니다.	아니요
`X-Real-IP`	CloudFront에서 헤더를 제거합니다.	아니요

HTTP 버전

CloudFront에서는 HTTP/1.1을 사용하여 사용자 지정 오리진에 요청을 전달합니다.

요청의 최대 길이 및 최대 URL 길이

경로, 쿼리 문자열(있는 경우), 헤더를 모두 포함한 요청의 최대 길이는 20,480바이트입니다.

CloudFront에서는 이 요청으로부터 URL을 구성합니다. 이 URL의 최대 길이는 8,192바이트입니다.

요청 또는 URL이 이 최대값을 초과할 경우 CloudFront에서는 HTTP 요청 코드 413(요청 개체가 너무 큼)을 반환한 후 최종 사용자와의TCP 연결을 종료합니다.

OCSP 스테이플링

최종 사용자가 객체에 대한 HTTPS 요청을 제출할 때 CloudFront 또는 최종 사용자는 CA(인증 기관)을 통해 도메인의 SSL 인증서가 해지되지 않았는지 확인해야 합니다. OCSP 스테이플링은 CloudFront에서 인증서의 유효성을 검사하고 CA로부터 응답을 캐싱할 수 있도록 함으로써 클라이언트가 CA를 통해 직접 인증서의 유효성을 검사하지 않아도 되므로 인증서 유효성 검사 속도가 향상됩니다.

OSCP 스테이플링의 성능 개선은 CloudFront에서 같은 도메인 내의 객체에 대해 많은 HTTPS 요청을 받은 경우 더욱 확연히 드러납니다. CloudFront 엣지 로케이션의 각 서버는 개별적인 유효성 검사 요청을 제출해야 합니다. CloudFront에서 같은 도메인에 대해 다수의 HTTPS 요청을 받은 경우, 엣지 로케이션의 각 서버에서는 곧 CA로부터 SSL 핸드셰이크의 패킷에 "스테이플"할 수 있다는 응답을 받습니다. 최종 사용자가 인증서가 유효하다는 조건을 충족하면 CloudFront에서는 요청된 객체를 제공할 수 있습니다. 배포가 CloudFront 엣지 로케이션에서 많은 양의 트래픽을 받지 않는 경우, 새로운 요청은 아직 CA를 통해 인증서의 유효성을 검사하지 않은 서버로 리디렉션될 가능성이 높습니다. 그러한 경우, 최종 사용자는 유효성 검사 단계를 개별적으로 수행하고 CloudFront 서버에서는 객체를 제공합니다. 해당 CloudFront 서버에서는 또한 유효성 검사 요청을 CA에 제출하고, 다음에 동일한 도메인 이름을 포함한 요청을 수신하면 CA로부터 유효성 검사 응답을 받습니다.

지속적인 연결

CloudFront가 오리진으로부터 응답을 받는 경우, 그 시간 동안 다른 요청이 도착하면 몇 초 정도 연결을 유지하려고 합니다. 지속적인 연결을 유지하면 TCP 연결 재설정 및 이후 요청에 대한 별도의 TLS 핸드쉐이크 수행에 필요한 시간이 절약됩니다.

지속적 연결의 지속 시간을 구성하는 방법 등 자세한 내용은 오리진 연결 유지 제한 시간 (p. 42) 단원의 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35)을 참조하십시오.

프로토콜

CloudFront에서는 다음 사항을 바탕으로 HTTP 또는 HTTPS 요청을 오리진 서버에 전달합니다.

- 최종 사용자가 CloudFront에 보낸 요청의 프로토콜(HTTP 또는 HTTPS)
- CloudFront 콘솔에서 오리진 프로토콜 정책 필드의 값, 또는 CloudFront API를 사용 중인 경우 `DistributionConfig` 복합 형식의 `OriginProtocolPolicy`, 요소. CloudFront 콘솔에는 HTTP Only(HTTP만 해당), HTTPS Only(HTTPS만 해당) 및 Match Viewer 옵션이 있습니다.

HTTP Only(HTTP만 해당) 또는 HTTPS Only(HTTPS만 해당)를 지정하는 경우, 최종 사용자 요청의 프로토콜과 관계없이 CloudFront에서는 지정된 프로토콜을 사용하여 오리진 서버로 요청을 전달합니다.

Match Viewer를 지정하는 경우, CloudFront에서는 최종 사용자 요청의 프로토콜을 사용하여 오리진 서버에 요청을 전달합니다. 최종 사용자가 HTTP 및 HTTPS 프로토콜 모두를 사용하여 요청하더라도 CloudFront에서는 한 번만 객체를 캐싱합니다.

> **Important**
>
> CloudFront가 HTTPS 프로토콜을 사용하여 원본으로 요청을 전달하고 오리진 서버가 잘못된 인증서 또는 자체 서명한 인증서를 반환하는 경우, CloudFront에서는 TCP 연결을 끊습니다.

CloudFront 콘솔을 사용하여 배포를 업데이트하는 방법에 대한 자세한 내용은 배포 업데이트 (p. 59) 단원을 참조하십시오. CloudFront API를 사용한 배포 업데이트 방법에 대한 자세한 내용은 Amazon CloudFront API Reference의 UpdateDistribution 단원을 참조하십시오.

쿼리 문자열

CloudFront에서 쿼리 문자열 파라미터를 오리진에 전달할지 여부를 구성할 수 있습니다. 자세한 내용은 쿼리 문자열 파라미터 기반의 콘텐츠 캐싱 (p. 196) 단원을 참조하십시오.

오리진 응답 제한 시간

오리진 응답 제한 시간(오리진 요청 제한 시간 또는 오리진 읽기 제한 시간이라고도 함)은 다음 두 값에 모두 적용됩니다.

- CloudFront가 사용자 지정 오리진에 요청을 전달한 후 응답을 기다리는 시간(초)
- CloudFront가 오리진으로부터 응답 패킷을 수신한 후 다음 패킷을 수신할 때까지 대기하는 시간(초)

오리진 응답 제한 시간을 구성하는 방법 등 자세한 내용은 오리진 응답 제한 시간 (p. 42) 단원의 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35)을 참조하십시오.

동일 객체에 대한 동시 요청(트래픽 스파이크)

CloudFront 엣지 로케이션에서 객체에 대한 요청을 받을 때 객체가 현재 캐시에 있지 않거나 객체가 만료된 경우, CloudFront에서는 즉시 요청을 오리진으로 보냅니다. 트래픽 스파이크가 있는 경우(오리진에서 첫 번째 요청에 응답하기 전에 동일 객체에 대한 추가 요청이 엣지 로케이션에 도착하는 경우) CloudFront에서는 객체에 대한 추가 요청을 오리진에 전달하기 전에 작업을 일시 중단합니다. 일반적으로는 후속 요청에 응답하기 전에 첫 번째 요청에 대한 응답이 CloudFront 엣지 로케이션에 도착하게 됩니다. 이러한 일시 중단은 오리진 서버에 불필요하게 로드가 걸리는 것을 줄여 줍니다. 요청 헤더나 쿠키에 따라 캐싱하도록 CloudFront를 구성하는 등의 이유로 추가 요청이 동일하지 않은 경우, CloudFront에서는 모든 고유한 요청을 오리진에 전달합니다.

User-Agent 헤더

CloudFront에서 사용자가 콘텐츠를 보기 위해 사용 중인 디바이스에 따라 여러 버전의 객체를 캐싱하도록 하려는 경우, 다음과 같이 하나 이상의 헤더를 사용자 지정 오리진에 전달하도록 CloudFront를 구성하는 것이 좋습니다.

- `CloudFront-Is-Desktop-Viewer`
- `CloudFront-Is-Mobile-Viewer`
- `CloudFront-Is-SmartTV-Viewer`
- `CloudFront-Is-Tablet-Viewer`

`User-Agent` 헤더의 값에 따라 CloudFront에서는 요청을 오리진에 전달하기 전에 이러한 헤더의 값을 `true` 또는 `false`로 설정합니다. 디바이스가 둘 이상의 범주에 해당하는 경우 둘 이상의 값이 `true`일 수 있습니다. 예를 들어, 일부 태블릿 디바이스의 경우 CloudFront에서는 `CloudFront-Is-Mobile-Viewer`와 `CloudFront-Is-Tablet-Viewer`를 모두 `true`로 설정할 수 있습니다. 요청 헤더에 따라 캐싱하도록 CloudFront를 구성하는 방법에 대한 자세한 내용은 요청 헤더에 따라 콘텐츠 캐싱 (p. 200) 단원을 참조하십시오.

`User-Agent` 헤더의 값에 따라 객체를 캐싱하도록 CloudFront를 구성할 수 있지만, 권장되지는 않습니다. `User-Agent` 헤더는 여러 가지 값을 가질 수 있으며, 그러한 값에 따른 캐싱으로 인해 CloudFront에서 오리진에 전달되는 요청이 눈에 띄게 증가할 수 있습니다.

`User-Agent` 헤더의 값에 따라 객체를 캐싱하도록 CloudFront를 구성하지 않는 경우, CloudFront에서는 오리진에 요청을 전달하기 전에 다음 값으로 `User-Agent` 헤더를 추가합니다.

```
User-Agent = Amazon CloudFront
```

CloudFront에서는 최종 사용자의 요청에 `User-Agent` 헤더가 포함되어 있는지 여부와 관계없이 이 헤더를 추가합니다. 최종 사용자의 요청에 `User-Agent` 헤더가 포함되어 있는 경우, CloudFront에서는 이를 제거합니다.

CloudFront에서 사용자 지정 오리진 서버의 요청을 처리하는 방법

이 주제에는 CloudFront가 사용자 지정 오리진에서 요청을 처리하는 방법에 대한 자세한 내용이 포함되어 있습니다.

주제

- 100-계속 응답 (p. 245)
- 캐싱 (p. 245)

- 취소된 요청 (p. 245)
- 콘텐츠 협상 (p. 245)
- 쿠키 (p. 245)
- TCP 연결 끊김 (p. 245)
- CloudFront에서 제거하거나 대체하는 HTTP 응답 헤더 (p. 246)
- 최대 파일 크기 (p. 246)
- 오리진 사용 불가 (p. 247)
- 리디렉션 (p. 247)
- 전송 인코딩 (p. 247)

100-계속 응답

오리진은 CloudFront에 하나 이상의 100-계속 응답을 전송할 수 없습니다. 첫 번째 100-계속 응답 후에 CloudFront는 HTTP 200 OK 응답을 예상합니다. 오리진이 첫 번째 응답 후 또 다른 100-계속 응답을 전송하면 CloudFront는 오류를 반환합니다.

캐싱

- 오리진 서버에서 `Date` 및 `Last-Modified` 헤더 필드에 유효하고 정확한 값을 설정했는지 확인합니다.
- 최종 사용자의 요청에 `If-Match` 또는 `If-None-Match` 요청 헤더 필드가 포함된 경우, `ETag` 응답 헤더 필드를 설정합니다. `ETag` 값을 지정하지 않은 경우, CloudFront에서는 후속 `If-Match` 또는 `If-None-Match` 헤더를 무시합니다.
- CloudFront에서는 오리진의 응답으로 보통 `Cache-Control: no-cache` 헤더를 신뢰합니다. 예외 사항은 동일 객체에 대한 동시 요청(트래픽 스파이크) (p. 244) 단원을 참조하십시오.

취소된 요청

객체가 엣지 캐시에 있지 않은 경우 최종 사용자가 CloudFront가 오리진에서 객체를 가져온 뒤 요청된 객체를 제공하기 전에 브라우저 닫기 등으로 세션을 종료하면, CloudFront에서는 엣지 로케이션의 객체를 캐싱하지 않습니다.

콘텐츠 협상

오리진이 응답에서 `Vary:*`를 반환하는 경우와 해당 캐시 동작의 최소 TTL 값이 0인 경우, CloudFront에서는 객체를 캐싱하지만 오리진에 객체의 모든 후속 요청을 전송하여 캐시에 객체의 최신 버전이 포함되어 있음을 확인합니다. CloudFront에는 `If-None-Match` 또는 `If-Modified-Since`와 같은 조건부 헤더가 포함되지 않습니다. 결과적으로 오리진은 모든 요청에 응답하여 객체를 CloudFront에 반환합니다.

오리진이 응답에서 `Vary:*`를 반환하는 경우와 해당 캐시 동작의 Minimum TTL 값이 다른 값인 경우, CloudFront에서는 CloudFront에서 제거하거나 대체하는 HTTP 응답 헤더 (p. 246)에 설명된 `Vary` 헤더를 처리합니다.

쿠키

캐시 동작에 대한 쿠키를 활성화한 경우 오리진에서 객체를 통해 쿠키를 반환하면 CloudFront에서는 객체와 쿠키를 모두 캐싱합니다. 이를 통해 객체에 대한 캐싱 가능성이 줄어듭니다. 자세한 내용은 쿠키를 기반으로 콘텐츠 캐싱 (p. 198) 단원을 참조하십시오.

TCP 연결 끊김

오리진이 객체를 CloudFront에 반환하는 동안 CloudFront와 오리진 간의 TCP 연결이 끊어진 경우, CloudFront 동작은 오리진이 응답에 `Content-Length` 헤더를 포함했는지 여부에 따라 달라집니다.

- Content-Length 헤더 – CloudFront에서는 오리진에서 객체를 가져오는 동안 최종 사용자에게 객체를 반환합니다. 그러나 `Content-Length` 헤더의 값이 객체의 크기와 일치하지 않는 경우, CloudFront에서는 객체를 캐싱하지 않습니다.
- Transfer-Encoding: Chunked – CloudFront에서는 오리진에서 객체를 가져오는 동안 최종 사용자에게 객체를 반환합니다. 그러나 조각난 응답이 완전하지 않은 경우, CloudFront에서는 객체를 캐싱하지 않습니다.
- No Content-Length 헤더 – CloudFront에서는 최종 사용자에게 객체를 반환하고 이를 캐싱하지만, 이 객체는 불완전할 수 있습니다. CloudFront에서는 `Content-Length` 헤더 없이는 TCP 연결이 실수로 또는 고의로 끊어졌는지 여부를 판단할 수 없습니다.

`Content-Length` 헤더를 추가하여 CloudFront에서 부분적인 객체를 캐싱하지 못하도록 HTTP 서버를 구성하는 것이 좋습니다.

CloudFront에서 제거하거나 대체하는 HTTP 응답 헤더

CloudFront는 오리진에서 최종 사용자에게 응답을 전달하기 전에 다음 헤더 필드를 제거하거나 업데이트합니다.

- `Set-Cookie` – 쿠키를 전달하도록 CloudFront를 구성하는 경우, `Set-Cookie` 헤더 필드가 클라이언트에 전달됩니다. 자세한 내용은 쿠키를 기반으로 콘텐츠 캐싱 (p. 198) 단원을 참조하십시오.
- `Trailer`
- `Transfer-Encoding` – 오리진에서 이 헤더 필드를 반환하는 경우, CloudFront에서는 최종 사용자에게 응답을 반환하기 전에 `chunked`에 값을 설정합니다.
- `Upgrade`
- `Vary` – 다음을 참조하십시오.
  - 디바이스별 헤더 중 하나를 오리진(`CloudFront-Is-Desktop-Viewer`, `CloudFront-Is-Mobile-Viewer`, `CloudFront-Is-SmartTV-Viewer`, `CloudFront-Is-Tablet-Viewer`)에 전달하도록 CloudFront를 구성하고 `Vary:User-Agent`를 CloudFront에 반환하도록 오리진을 구성하는 경우, CloudFront에서는 최종 사용자에게 `Vary:User-Agent`를 반환합니다. 자세한 내용은 디바이스 유형을 기반으로 캐싱하도록 구성 (p. 203) 단원을 참조하십시오.
  - `Vary` 헤더에서 `Accept-Encoding` 또는 `Cookie`를 포함하도록 오리진을 구성하는 경우, CloudFront는 최종 사용자에 대한 응답에 대한 값을 포함합니다.
  - 헤더 화이트리스트를 오리진에 전달하도록 CloudFront를 구성하는 경우와 `Vary` 헤더(예를 들어 `Vary:Accept-Charset,Accept-Language`)의 CloudFront에 헤더 이름을 반환하도록 오리진을 구성하는 경우, CloudFront에서는 최종 사용자에게 `Vary` 헤더를 그 값과 함께 반환합니다.
  - CloudFront가 `Vary` 헤더의 `*` 값을 처리하는 방법에 대한 자세한 내용은 콘텐츠 협상 (p. 245) 단원을 참조하십시오.
  - `Vary` 헤더에 그 밖의 값을 포함하도록 오리진을 구성하는 경우, CloudFront는 최종 사용자에게 응답을 반환하기 전에 값을 제거합니다.
- `Via` – CloudFront는 최종 사용자에 대한 응답으로 값을 다음과 같이 설정합니다.

  ```
  Via: http-version alphanumeric-string.cloudfront.net (CloudFront)
  ```

  예를 들어, 클라이언트가 HTTP/1.1을 통해 요청한 경우 값은 다음과 같습니다.

  ```
  Via: 1.1 1026589cc7887e7a0dc7827b4example.cloudfront.net (CloudFront)
  ```

최대 파일 크기

CloudFront에서 최종 사용자에게 반환하는 응답 본문의 최대 크기는 20GB입니다. 여기에는 `Content-Length` 헤더 값으로 지정하지 않은 조각난 전송 응답이 포함됩니다.

오리진 사용 불가

오리진 서버를 사용할 수 없고 CloudFront에서 엣지 캐시에 있지만 만료된(`Cache-Control max-age` 명령에 지정된 기간이 지났다는 이유 등으로) 객체에 대한 요청을 받은 경우, CloudFront에서는 만료된 버전의 객체를 제공하거나 사용자 지정 오류 페이지를 표시합니다. 사용자 지정 오류 페이지를 구성한 경우 CloudFront 동작에 대한 자세한 내용은 사용자 지정 오류 페이지를 구성했을 때 CloudFront에서 오류를 처리하는 방법 (p. 251) 단원을 참조하십시오.

경우에 따라 자주 요청되지는 않는 객체는 제거되고 엣지 캐시에서 더 이상 사용할 수 없게 됩니다. CloudFront에서는 제거된 객체를 제공할 수 없습니다.

리디렉션

오리진 서버에 있는 객체의 위치를 변경하는 경우 요청을 새 위치로 리디렉션하도록 웹 서버를 구성할 수 있습니다. 리디렉션을 구성한 뒤 처음으로 최종 사용자가 객체에 대한 요청을 제출할 때 CloudFront에서는 요청을 오리진에 전송하고 오리진에서는 리디렉션으로 응답합니다(예: `302 Moved Temporarily`). CloudFront에서는 이 리디렉션을 캐싱하고 최종 사용자에게 이를 반환합니다. CloudFront에서는 이 리디렉션을 따라가지 않습니다.

다음 위치 중 하나로 요청을 리디렉션하도록 웹 서버를 구성할 수 있습니다.

- 오리진 서버에 있는 객체의 새 URL입니다. 최종 사용자가 새 URL에 대한 리디렉션을 따라갈 때 최종 사용자는 CloudFront를 우회하고 오리진으로 직행합니다. 따라서 오리진에 있는 객체의 새 URL로 요청을 리디렉션하지 않는 것이 좋습니다.
- 객체에 대한 새 CloudFront URL입니다. 최종 사용자가 새 CloudFront URL을 포함한 요청을 제출할 때, CloudFront는 새 위치에서 객체를 가져오고 이를 엣지 로케이션에서 캐싱하여 최종 사용자에게 반환합니다. 이 객체에 대한 후속 요청은 엣지 로케이션에서 제공됩니다. 이를 통해 최종 사용자의 오리진에서의 객체 요청과 관련된 시간 지연과 로드 발생을 피할 수 있습니다. 그러나 객체에 대한 새로운 요청이 발생할 때마다 CloudFront에 대한 두 개의 요청에 대해 요금이 부과됩니다.

전송 인코딩

CloudFront에서는 `Transfer-Encoding` 헤더에 대해 `chunked` 값만 지원합니다. 오리진에서 `Transfer-Encoding: chunked`를 반환하는 경우, CloudFront에서는 객체를 엣지 로케이션에서 수신한 객체로 클라이언트에 반환하고 후속 요청에 대해 chunked 형식의 객체를 캐싱합니다.

최종 사용자가 `Range GET` 요청을 하고 오리진이 `Transfer-Encoding: chunked`를 반환하는 경우, CloudFront에서는 요청한 범위 대신에 뷰어에 전체 객체를 반환합니다.

응답의 콘텐츠 길이를 사전에 결정할 수 없는 경우 chunked 인코딩을 사용하는 것이 좋습니다. 자세한 내용은 TCP 연결 끊김 (p. 245) 단원을 참조하십시오.

오리진 그룹에 대한 요청 및 응답 동작

오리진 그룹에 대한 요청은, 오리진 장애 조치를 유발하는 캐시 누락이 있는 경우를 제외하면, 오리진 그룹으로 설정되지 않은 오리진에 대한 응답과 유사하게 작동합니다. 다른 모든 오리진에서와 마찬가지로, CloudFront가 요청을 수신하고 콘텐츠가 이미 엣지 로케이션에 캐시되는 경우, 콘텐츠는 캐시로부터 최종 사용자에게 제공됩니다. 오리진 장애 조치를 설정했고 캐시 누락이 있는 경우, 최종 사용자의 요청은 오리진 그룹의 기본 오리진으로 전달됩니다.

기본 오리진에 대한 요청 및 응답 동작은 오리진 그룹에 포함되지 않은 오리진에 대한 경우와 동일합니다. 자세한 내용은 Amazon S3 오리진에 대한 요청 및 응답 동작 (p. 230) 및 사용자 지정 오리진에 대한 요청 및 응답 동작 (p. 236) 단원을 참조하십시오.

다음은 기본 오리진이 특정 HTTP 상태 코드를 반환할 때의 오리진 장애 조치에 대한 동작에 대한 설명입니다.

- HTTP 2xx 상태 코드(성공): CloudFront가 파일을 캐싱하여 최종 사용자에게 반환합니다.
- HTTP 3xx 상태 코드(리디렉션): CloudFront가 상태 코드를 최종 사용자에게 반환합니다.
- HTTP 4xx 또는 5xx 상태 코드(클라이언트/서버 오류): 반환된 상태 코드가 장애 조치에 대하여 구성된 경우, CloudFront가 오리진 그룹의 두 번째 오리진으로 전환하여 파일을 요청합니다.
- HTTP 4xx 또는 5xx 상태 코드(클라이언트/서버 오류): 반환된 상태 코드가 장애 조치에 대하여 구성된 경우, CloudFront가 오류를 최종 사용자에게 반환합니다.

CloudFront가 두 번째 오리진에 파일을 요청하는 경우, 응답 동작은 오리진 그룹에 설정되지 않은 CloudFront 오리진에 대한 경우와 동일합니다.

오리진 그룹에 대한 자세한 내용은 CloudFront 오리진 장애 조치를 통한 고가용성 최적화 (p. 209) 단원을 참조하십시오.

사용자 지정 헤더를 오리진 요청에 추가

오리진으로 보내는 요청에 사용자 지정 헤더를 추가하도록 CloudFront를 구성할 수 있습니다. 이러한 사용자 지정 헤더를 사용하면 일반적인 최종 사용자 요청을 통해 얻을 수 없는 오리진에서 정보를 보내고 수집할 수 있습니다. 각 오리진에 대해 이러한 헤더를 사용자 지정할 수도 있습니다. CloudFront는 사용자 지정 오리진 및 Amazon S3 오리진 모두에 사용자 지정 헤더를 지원합니다.

> **Note**
>
> 사용자 지정 헤더의 추가는 웹 배포에만 적용됩니다.

주제

- 오리진 사용자 지정 헤더의 사용 사례 (p. 248)
- 사용자 지정 헤더를 오리진 요청에 추가하도록 CloudFront 구성 (p. 249)
- CloudFront가 오리진 요청에 추가할 수 없는 사용자 지정 헤더 (p. 249)
- 권한 부여 헤더를 전달하도록 CloudFront 구성 (p. 250)

오리진 사용자 지정 헤더의 사용 사례

다음과 같이 여러 경우에 사용자 지정 헤더를 사용할 수 있습니다.

CloudFront에서 요청 식별

오리진이 CloudFront에서 수신한 요청을 식별할 수 있습니다. 사용자가 CloudFront를 우회하는지 알고 싶은 경우나 2개 이상의 CDN을 사용하고 있는데 어떤 요청이 각 CDN에서 오는지에 대한 정보를 원하는 경우 유용합니다.

> **Note**
>
> Amazon S3 오리진을 사용하고 Amazon S3 서버 액세스 로깅을 활성화하는 경우, 로그에는 헤더 정보가 포함되지 않습니다.

어떤 요청이 특정 배포에서 오는지 확인

동일한 오리진을 사용하도록 둘 이상의 CloudFront 배포를 구성하는 경우, 각 배포에 각각 다른 사용자 지정 헤더를 추가할 수 있습니다. 그러면 오리진의 로그를 사용하여 어떤 요청이 어떤 CloudFront 배포에서 왔는지 확인할 수 있습니다.

Cross-Origin 리소스 공유(CORS) 활성화

일부 최종 사용자가 Cross-Origin 리소스 공유(CORS)를 지원하지 않는 경우, 오리진으로 보내는 요청에 `Origin` 헤더를 항상 추가하도록 CloudFront를 구성할 수 있습니다. 그러면 모든 요청에 대해 `Access-Control-Allow-Origin` 헤더를 반환하도록 오리진을 구성할 수 있습니다. 또한 CORS 설정을 지키도록 CloudFront를 구성 (p. 202)해야 합니다.

콘텐츠에 대한 액세스 제어

사용자 지정 헤더를 사용하여 콘텐츠에 대한 액세스를 제어할 수 있습니다. CloudFront에 의해 추가된 사용자 지정 헤더가 포함된 경우에만 요청에 응답하도록 오리진을 구성하여 사용자가 CloudFront를 우회하는 것과 오리진에서 직접 콘텐츠에 액세스하는 것을 방지할 수 있습니다. 자세한 내용은 사용자 지정 오리진의 파일에 대한 액세스 제한 (p. 117) 단원을 참조하십시오.

사용자 지정 헤더를 오리진 요청에 추가하도록 CloudFront 구성

오리진으로 보내는 요청에 사용자 지정 헤더를 추가하도록 웹 배포를 구성하려면 다음 방법 중 하나를 사용하여 오리진의 구성을 업데이트합니다.

- CloudFront 콘솔 – 배포를 생성 또는 업데이트할 경우, 오리진 사용자 지정 헤더 설정에 헤더 이름과 값을 지정합니다. 자세한 내용은 배포 만들기 (p. 35) 또는 배포 업데이트 (p. 59) 단원을 참조하십시오.
- CloudFront API – 사용자 지정 헤더를 추가하려는 각 오리진에 대해 `CustomHeaders` 필드 내부 `Origin`에 헤더 이름과 값을 지정합니다. 자세한 내용은 CreateDistribution 또는 UpdateDistribution을 참조하십시오.

지정하는 헤더 이름과 값이 아직 최종 사용자 요청에 있지 않은 경우, CloudFront는 이를 오리진 요청에 추가합니다. 헤더가 있는 경우, CloudFront는 오리진에 요청을 전달하기 전에 헤더 값을 덮어씁니다.

오리진 사용자 지정 헤더에 적용되는 할당량(이전에는 제한이라고 함)에 대한 자세한 내용은 사용자 지정 헤더에 대한 할당량(웹 배포만 해당) (p. 449) 단원을 참조하십시오.

CloudFront가 오리진 요청에 추가할 수 없는 사용자 지정 헤더

오리진으로 보내는 요청에 다음 헤더 중 하나라도 추가하도록 CloudFront를 구성할 수 없습니다.

- `Cache-Control`
- `Connection`
- `Content-Length`
- `Cookie`
- `Host`
- `If-Match`
- `If-Modified-Since`
- `If-None-Match`
- `If-Range`
- `If-Unmodified-Since`
- `Max-Forwards`
- `Pragma`
- `Proxy-Authorization`
- `Proxy-Connection`

- `Range`
- `Request-Range`
- `TE`
- `Trailer`
- `Transfer-Encoding`
- `Upgrade`
- `Via`
- `X-Amz-`로 시작되는 헤더
- `X-Edge-`로 시작되는 헤더
- `X-Real-Ip`

권한 부여 헤더를 전달하도록 CloudFront 구성

요청을 오리진으로 전달하면 CloudFront는 기본적으로 권한 부여 헤더를 포함하여 일부 최종 사용자를 제거합니다. 특정 캐시 동작에 대해 항상 권한 부여 헤더를 전달하려면 해당 캐시 동작에 대해 헤더 화이트리스트를 작성합니다. 자세한 내용은 Whitelist Headers (p. 47) 단원을 참조하십시오.

CloudFront에서 오리진 서버의 HTTP 3xx 상태 코드를 처리하는 방법

CloudFront가 Amazon S3 버킷이나 사용자 지정 오리진 서버에서 객체를 요청하는 경우 때때로 오리진은 HTTP 3xx 상태 코드를 반환하는데 이것은 보통 URL이 변경되었음(301, 영구적으로 이동됨 또는 307, 임시 리디렉션) 또는 객체가 CloudFront가 마지막으로 요청한 이후 변경되지 않았음(304, 수정되지 않음)을 나타냅니다. CloudFront는 CloudFront 배포의 설정과 오리진이 객체와 함께 반환되는 헤더 필드에 의해 지정된 기간 동안 3xx 응답을 캐싱합니다. 자세한 내용은 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204) 단원을 참조하십시오.

오리진이 301 또는 307 상태 코드를 반환하는 경우 CloudFront는 새 로케이션에 대한 리디렉션을 따라가지 않습니다.

CloudFront에서 오리진의 HTTP 4xx 및 5xx 상태 코드를 처리 및 캐싱하는 방법

주제

- 사용자 지정 오류 페이지를 구성했을 때 CloudFront에서 오류를 처리하는 방법 (p. 251)
- 사용자 지정 오류 페이지를 구성하지 않았을 때 CloudFront에서 오류를 처리하는 방법 (p. 252)
- CloudFront에서 캐싱하는 HTTP 4xx 및 5xx 상태 코드 (p. 253)

CloudFront가 Amazon S3 버킷 또는 사용자 지정 오리진 서버에서 객체를 요청할 때 오리진에서 오류가 발생했음을 나타내는 HTTP 4xx 또는 5xx 상태 코드를 반환하는 경우가 있습니다. CloudFront 동작은 다음에 따라 달라집니다.

- 사용자 지정 오류 페이지를 구성했는지 여부
- CloudFront에서 오리진으로부터 오류 응답을 캐싱하려는 시간(오류 캐싱 최소 TTL)을 구성했는지 여부
- 상태 코드

- 5xx 상태 코드의 경우 요청된 객체가 현재 CloudFront 엣지 캐시에 있는지 여부
- 일부 4xx 상태 코드의 경우 오리진에서 `Cache-Control max-age` 또는 `Cache-Control s-maxage` 헤더를 반환하는지 여부

CloudFront에서는 `GET` 및 `HEAD` 요청에 대한 응답을 항상 캐싱합니다. 또한 `OPTIONS` 요청에 대한 응답을 캐싱하도록 CloudFront를 구성할 수도 있습니다. CloudFront에서는 다른 메서드를 사용하는 요청에 대한 응답을 캐싱하지 않습니다.

오리진이 응답하지 않는 경우 오리진에 대한 CloudFront 요청은 시간 초과됩니다. 이 상태는 오리진이 해당 오류로 응답하지 않았더라도 오리진의 HTTP 5xx 오류로 간주됩니다. 해당 시나리오에서 CloudFront는 캐시된 콘텐츠를 제공합니다. 자세한 내용은 오리진 사용 불가 (p. 247) 단원을 참조하십시오.

로깅을 활성화한 경우 CloudFront에서는 HTTP 상태 코드에 관계없이 결과를 로그에 작성합니다.

CloudFront에서 반환되는 오류 메시지에 관련된 기능 및 옵션에 대한 자세한 내용은 다음을 참조하십시오.

- CloudFront 콘솔에서 사용자 지정 오류 페이지를 설정하는 방법에 대한 자세한 내용은 사용자 지정 오류 페이지 및 오류 캐싱 (p. 56) 단원을 참조하십시오.
- CloudFront 콘솔의 오류 캐싱 최소 TTL에 관한 자세한 내용은 Error Caching Minimum TTL (p. 57) 단원을 참조하십시오.
- CloudFront에서 캐싱하는 HTTP 상태 코드의 목록은 CloudFront에서 캐싱하는 HTTP 4xx 및 5xx 상태 코드 (p. 253) 단원을 참조하십시오.

사용자 지정 오류 페이지를 구성했을 때 CloudFront에서 오류를 처리하는 방법

사용자 지정 오류 페이지를 구성한 경우 CloudFront의 동작은 요청된 객체가 엣지 캐시에 있는지 여부에 따라 달라집니다.

요청된 객체가 엣지 캐시에 없는 경우

CloudFront는 다음이 모두 참이면 오리진에서 요청한 객체를 계속 가져오려 합니다.

- 최종 사용자가 객체를 요청하는 경우
- 객체가 엣지 캐시에 있지 않는 경우
- 오리진이 HTTP 4xx 또는 5xx 상태 코드를 반환하며 다음 중 하나가 true인 경우
  - 오리진이 304 상태 코드(수정되지 않음) 또는 객체의 업데이트된 버전을 반환하는 대신에 HTTP 5xx 상태 코드를 반환하는 경우
  - 오리진이 캐시 제어 헤더에 의해 제한되지 않으며 상태 코드 목록(CloudFront에서 항상 캐시하는 HTTP 4xx 및 5xx 상태 코드 (p. 253))에 포함된 HTTP 4xx 상태 코드를 반환하는 경우
  - 오리진이 `Cache-Control max-age` 헤더 또는 `Cache-Control s-maxage` 헤더 없이 HTTP 4xx 상태 코드를 반환하며 상태 코드가 Control 캐시 제어 헤더에 따라 CloudFront에서 캐시하는 HTTP 4xx 상태 코드 (p. 254)의 상태 코드 목록에 포함된 경우

CloudFront는 다음 동작을 수행합니다.

1. 최종 사용자 요청을 받은 CloudFront 엣지 캐시에서 CloudFront는 배포 구성을 확인하여 오리진 서버에서 반환한 상태 코드에 해당하는 사용자 지정 오류 페이지의 경로를 가져옵니다.
2. CloudFront에서는 사용자 지정 오류 페이지의 경로와 일치하는 경로 패턴을 가진 배포의 첫 번째 캐시 동작을 확인합니다.
3. CloudFront 엣지 로케이션에서는 캐시 동작에 지정된 오리진에 사용자 지정 오류 페이지에 대한 요청을 전송합니다.

4. 오리진에서는 엣지 로케이션에 사용자 지정 오류 페이지를 반환합니다.
5. CloudFront가 요청한 최종 사용자에게 사용자 지정 오류 페이지를 반환하며, 다음의 최대값에 대해 사용자 지정 오류 페이지도 캐시합니다.
   - 오류 캐싱 최소 TTL(기본값: 5분)에 의해 지정된 시간
   - 첫 번째 요청이 오류를 생성할 때 오리진에서 반환된 `Cache-Control max-age` 헤더 또는 `Cache-Control s-maxage` 헤더에 의해 지정된 시간
6. 캐시 시간(5단계에서 결정)이 경과한 후, CloudFront에서는 다른 요청을 오리진에 전달함으로써 요청된 객체를 다시 가져오려고 시도합니다. CloudFront는 오류 캐시 최소 TTL에 의해 지정된 간격으로 계속 다시 시도합니다.

요청된 객체가 엣지 캐시에 있는 경우

CloudFront는 다음이 모두 참이면 현재 엣지 캐시에 있는 객체를 계속 제공하려 합니다.

- 최종 사용자가 객체를 요청하는 경우
- 객체가 엣지 캐시에 있지만 만료된 경우
- 오리진이 304 상태 코드(수정되지 않음) 또는 객체의 업데이트된 버전을 반환하는 대신에 HTTP 5xx 상태 코드를 반환하는 경우

CloudFront는 다음 동작을 수행합니다.

1. 오리진이 5xx 상태 코드를 반환할 경우 CloudFront에서는 객체가 만료되었더라도 이를 제공합니다. 오류 캐싱 최소 TTL 동안 CloudFront는 엣지 캐시에서 객체를 제공함으로써 최종 사용자 요청에 지속적으로 응답합니다.

   오리진이 4xx 상태 코드를 반환할 경우 CloudFront는 최종 사용자에게 요청된 객체가 아니라 상태 코드를 반환합니다.
2. 오류 캐싱 최소 TTL이 경과한 후, CloudFront에서는 다른 요청을 오리진에 전달함으로써 요청된 객체를 다시 가져오려고 시도합니다. 객체가 자주 요청되지 않는 경우, CloudFront에서는 오리진 서버가 아직 5xx 응답을 반환하는 중이라도 이를 엣지 캐시에서 제거할 수 있습니다. 객체가 CloudFront 엣지 캐시에 보관되는 시간에 대한 자세한 내용은 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204) 단원을 참조하십시오.

사용자 지정 오류 페이지를 구성하지 않았을 때 CloudFront에서 오류를 처리하는 방법

사용자 지정 오류 페이지를 구성하지 않은 경우 CloudFront의 동작은 요청된 객체가 엣지 캐시에 있는지 여부에 따라 달라집니다.

요청된 객체가 엣지 캐시에 없는 경우

CloudFront는 다음이 모두 참이면 오리진에서 요청한 객체를 계속 가져오려 합니다.

- 최종 사용자가 객체를 요청하는 경우
- 객체가 엣지 캐시에 있지 않는 경우
- 오리진이 HTTP 4xx 또는 5xx 상태 코드를 반환하며 다음 중 하나가 true인 경우
  - 오리진이 304 상태 코드(수정되지 않음) 또는 객체의 업데이트된 버전을 반환하는 대신에 HTTP 5xx 상태 코드를 반환하는 경우
  - 오리진이 캐시 제어 헤더에 의해 제한되지 않으며 상태 코드 목록(CloudFront에서 항상 캐시하는 HTTP 4xx 및 5xx 상태 코드 (p. 253))에 포함된 HTTP 4xx 상태 코드를 반환하는 경우

- 오리진이 `Cache-Control max-age` 헤더 또는 `Cache-Control s-maxage` 헤더 없이 HTTP 4xx 상태 코드를 반환하며 상태 코드가 Control 캐시 제어 헤더에 따라 CloudFront에서 캐시하는 HTTP 4xx 상태 코드 (p. 254)의 상태 코드 목록에 포함된 경우

CloudFront는 다음 동작을 수행합니다.

1. CloudFront는 최종 사용자에게 4xx 또는 5xx 상태 코드를 반환하며, 다음의 최대값에 대해 요청을 수신하는 엣지 캐시에 상태 코드도 캐시합니다.
   - 오류 캐싱 최소 TTL(기본값: 5분)에 의해 지정된 시간
   - 첫 번째 요청이 오류를 생성할 때 오리진에서 반환된 `Cache-Control max-age` 헤더 또는 `Cache-Control s-maxage` 헤더에 의해 지정된 시간
2. 캐시 시간(1단계에서 결정됨) 동안 CloudFront에서는 동일 객체에 대한 후속 최종 사용자 요청에 캐시된 4xx 또는 5xx 상태 코드로 응답합니다.
3. 캐시 시간(1단계에서 결정)이 경과한 후, CloudFront에서는 다른 요청을 오리진에 전달함으로써 요청된 객체를 다시 가져오려고 시도합니다. CloudFront는 오류 캐시 최소 TTL에 의해 지정된 간격으로 계속 다시 시도합니다.

요청된 객체가 엣지 캐시에 있는 경우

CloudFront는 다음이 모두 참이면 현재 엣지 캐시에 있는 객체를 계속 제공하려 합니다.

- 최종 사용자가 객체를 요청하는 경우
- 객체가 엣지 캐시에 있지만 만료된 경우
- 오리진이 304 상태 코드(수정되지 않음) 또는 객체의 업데이트된 버전을 반환하는 대신에 HTTP 5xx 상태 코드를 반환하는 경우

CloudFront는 다음 동작을 수행합니다.

1. 오리진이 5xx 오류 코드를 반환할 경우 CloudFront에서는 객체가 만료되었더라도 이를 제공합니다. 오류 캐싱 최소 TTL(기본 5분) 동안 CloudFront는 엣지 캐시에서 객체를 제공함으로써 최종 사용자 요청에 지속적으로 응답합니다.

   오리진이 4xx 상태 코드를 반환할 경우 CloudFront는 최종 사용자에게 요청된 객체가 아니라 상태 코드를 반환합니다.
2. 오류 캐싱 최소 TTL이 경과한 후, CloudFront에서는 다른 요청을 오리진에 전달함으로써 요청된 객체를 다시 가져오려고 시도합니다. 객체가 자주 요청되지 않는 경우, CloudFront에서는 오리진 서버가 아직 5xx 응답을 반환하는 중이라도 이를 엣지 캐시에서 제거할 수 있습니다. 객체가 CloudFront 엣지 캐시에 보관되는 시간에 대한 자세한 내용은 콘텐츠가 엣지 캐시에 유지되는 기간 관리(만료) (p. 204) 단원을 참조하십시오.

CloudFront에서 캐싱하는 HTTP 4xx 및 5xx 상태 코드

CloudFront에서는 반환된 특정 상태 코드 및 오리진이 응답으로 특정 헤더를 반환하는지 여부에 따라 오리진에서 반환된 HTTP 4xx 및 5xx 상태 코드를 캐시합니다.

CloudFront에서 항상 캐시하는 HTTP 4xx 및 5xx 상태 코드

CloudFront에서는 항상 오리진에서 반환한 다음 HTTP 4xx 및 5xx 상태 코드를 캐시합니다. HTTP 상태 코드에 대해 사용자 지정 오류 페이지를 구성한 경우, CloudFront에서는 사용자 지정 오류 페이지를 캐싱합니다.

404	찾을 수 없음

405	허용되지 않은 메소드
414	요청-URI가 너무 큼
500	내부 서버 오류
501	구현되지 않음
502	잘못된 게이트웨이
503	서비스 사용 불가
504	게이트웨이 시간 초과

캐시 제어 헤더에 따라 CloudFront에서 캐시하는 HTTP 4xx 상태 코드

CloudFront에서는 오리진이 `Cache-Control max-age` 또는 `Cache-Control s-maxage` 헤더를 반환할 경우 오리진에서 반환된 다음 HTTP 4xx 상태 코드만 캐시합니다. 이러한 HTTP 상태 코드 중 하나에 대해 사용자 지정 오류 페이지를 구성한 경우 오리진이 캐시 제어 헤더 중 하나를 반환하면 CloudFront는 사용자 지정 오류 페이지를 캐시합니다.

400	잘못된 요청
403	금지됨
412	사전 조건 실패
415	지원되지 않는 미디어 유형

사용자 지정 오류 응답 생성 중

CloudFront를 통해 제공하는 객체를 특정 이유로 인해 사용할 수 없는 경우, 웹 서버에서는 대개 HTTP 상태 코드를 CloudFront에 반환합니다. 예를 들어 최종 사용자가 잘못된 URL을 지정한 경우 웹 서버에서는 404 상태 코드를 CloudFront에 반환하고 CloudFront에서는 다시 최종 사용자에게 이 상태 코드를 반환합니다. 뷰어에는 다음과 유사한 짧고 불완전한 형식의 기본 메시지가 표시됩니다.

```
Not Found: The requested URL /myfilename.html was not found on this server.
```

하지만 원할 경우 대신 사용자 정의 오류 메시지를 표시할 수 있습니다. 또한 오류가 발생할 경우 CloudFront가 응답하는 방식을 관리할 수 있는 몇 가지 옵션이 제공됩니다. 사용자 지정 오류 메시지에 대한 옵션을 지정하려면 CloudFront 배포를 업데이트하여 해당 값을 지정합니다. 자세한 내용은 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 주제의 사용자 지정 오류 페이지 및 오류 캐싱 (p. 56) 단원을 참조하십시오.

주제

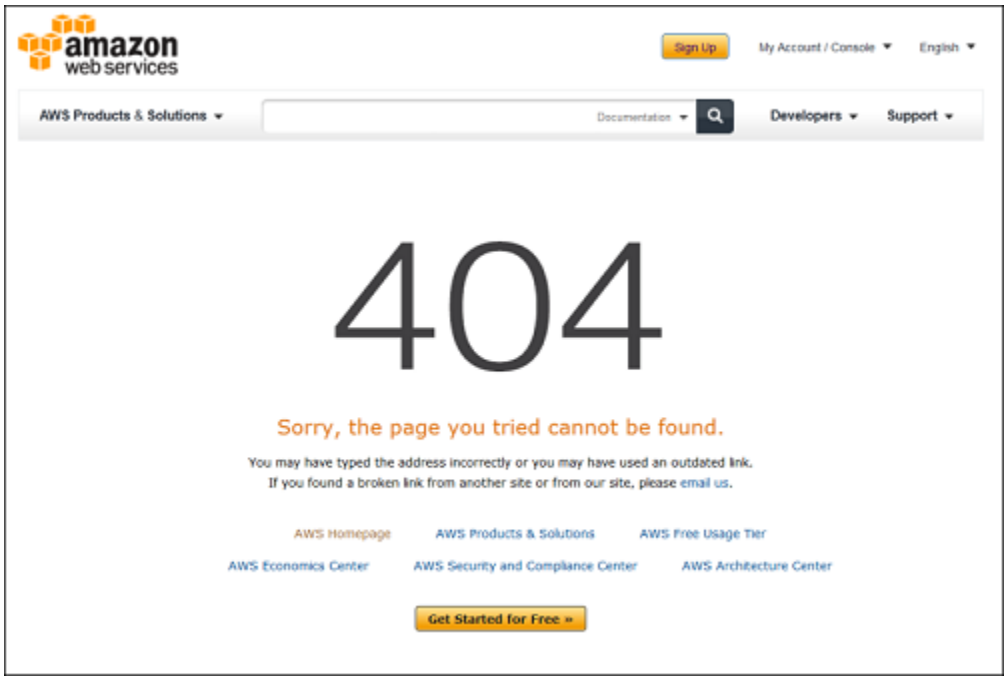
- 특정 HTTP 상태 코드에 대한 사용자 지정 오류 페이지 만들기 (p. 255)
- 다른 위치에 객체 및 사용자 지정 오류 페이지 저장 (p. 257)
- CloudFront에서 반환하는 응답 코드 변경 (p. 257)
- CloudFront에서 오류를 캐싱하는 기간 제어 (p. 257)
- 사용자 지정 오류 페이지를 사용할 수 없을 때 CloudFront에서 응답하는 방식 (p. 258)
- 사용자 지정 오류 페이지에 대한 요금 (p. 259)
- 오류 응답 동작 구성 (p. 259)

특정 HTTP 상태 코드에 대한 사용자 지정 오류 페이지 만들기

기본 메시지 대신 나머지 웹 사이트와 동일한 형식을 사용하는 페이지와 같은 사용자 지정 오류 메시지를 표시하려는 경우, CloudFront가 최종 사용자에게 사용자 지정 오류 메시지가 포함된 HTML 파일 등의 객체를 반환하게 하면 됩니다.

특정 파일을 반환해야 하는 오류 및 해당 파일을 지정하려면 CloudFront 배포를 업데이트하여 해당 값을 지정합니다. 자세한 내용은 사용자 지정 오류 페이지 및 오류 캐싱 (p. 56) 주제에서 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.

다음은 사용자 지정된 메시지의 예입니다.



지원되는 HTTP 상태 코드 각각에 서로 다른 객체를 지정하거나 지원되는 상태 코드 전체에 동일한 객체를 사용할 수 있습니다. 일부 상태 코드에 대해 객체를 지정하고 나머지에 대해서는 지정하지 않도록 선택할 수도 있습니다.

CloudFront를 통해 제공하는 객체는 다양한 이유로 사용 불가능할 수 있습니다. 이러한 이유는 넓게 보면 다음 두 가지 범주로 나뉩니다.

- 클라이언트 오류: 요청에 문제가 있음을 나타냅니다. 지정된 이름의 객체를 사용할 수 없거나, 사용자에게 Amazon S3 버킷의 객체를 가져오기 위해 필요한 권한이 없는 경우를 예로 들 수 있습니다. 클라이언트 오류가 발생하면 오리진에서는 400번대의 HTTP 상태 코드를 CloudFront에 반환합니다.
- 서버 오류: 오리진 서버에 문제가 있음을 나타냅니다. HTTP 서버가 사용 중이거나 사용 불가능한 상태인 경우를 예로 들 수 있습니다. 서버 오류가 발생하면 오리진 서버에서 500번대의 HTTP 상태 코드를 CloudFront에 반환합니다. 또는 특정 기간 동안 CloudFront에서 오리진 서버로부터 응답을 가져오지 않고 504 상태 코드(게이트웨이 시간 초과)를 적용합니다.

CloudFront에서 사용자 지정 오류 페이지를 반환할 수 있는 HTTP 상태 코드에는 다음이 포함됩니다.

- 400, 403, 404, 405, 414, 416
- 500, 501, 502, 503, 504

> **Note**
>
> HTTP 상태 코드 416(요청된 범위를 충족할 수 없음)에 대한 사용자 지정 오류 페이지를 만들 수 있으며 오리진에서 CloudFront로 상태 코드 416을 반환할 경우 CloudFront에서 최종 사용자로 반환하는 HTTP 상태 코드를 변경할 수 있습니다. (자세한 내용은 CloudFront에서 반환하는 응답 코드 변경 (p. 257) 단원을 참조하십시오.) 하지만 CloudFront에서는 상태 코드 416 응답을 캐싱하지 않으며, 따라서 상태 코드 416에 대해 Error Caching Minimum TTL(최소 TTL 캐싱 오류)의 값을 지정할 수 있지만 CloudFront에서는 이 값이 사용되지 않습니다.

CloudFront에서 오리진에서 보낸 오류 응답을 처리하는 방법에 대한 자세한 설명을 보려면 CloudFront에서 오리진의 HTTP 4xx 및 5xx 상태 코드를 처리 및 캐싱하는 방법 (p. 250) 단원을 참조하십시오.

다른 위치에 객체 및 사용자 지정 오류 페이지 저장

객체와 사용자 지정 오류 페이지를 서로 다른 위치에 저장하려는 경우, 배포에는 다음을 충족하는 캐시 동작이 포함되어야 합니다.

- 경로 패턴의 값이 사용자 지정 오류 메시지와 일치합니다. 예를 들어, `/4xx-errors`라는 이름의 디렉터리에 있는 Amazon S3 버킷에 4xx 오류에 대한 사용자 지정 오류 페이지를 저장했다고 가정합니다. 배포에는 사용자 지정 오류 페이지에 대한 요청을 해당 위치로 라우팅하는 경로 패턴(예: /4xx-errors/*)의 캐시 동작이 포함되어야 합니다.
- Origin(오리진)의 값은 사용자 지정 오류 페이지를 포함한 오리진에 대한 Origin ID(오리진 ID)의 값을 지정합니다.

자세한 내용은 캐시 동작 설정 (p. 43) 주제에서 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.

CloudFront에서 반환하는 응답 코드 변경

CloudFront에서 지정된 HTTP 상태 코드에 대해 사용자 지정 오류 페이지와 함께 반환하는 HTTP 상태 코드를 선택할 수 있습니다. 예를 들어 오리진에서 CloudFront에 500 상태 코드를 반환하는 경우, CloudFront에서 사용자 지정 오류 페이지와 200 상태 코드(OK)를 최종 사용자에게 반환하려 할 수 있습니다. CloudFront에서 오리진에서 최종 사용자에게 반환한 것과 다른 상태 코드를 CloudFront로 반환하려는 이유는 다음과 같이 다양합니다.

- 일부 인터넷 디바이스(예: 일부 방화벽 및 기업 프록시)에서는 HTTP 4xx 및 5xx를 가로채서 응답이 최종 사용자에게 반환되는 것을 막습니다. 이를 `200`으로 대체하면 이 응답은 일반적으로 가로채지 않습니다.
- 여러 가지 클라이언트 오류 또는 서버 오류 중에 구별할 필요가 없는 경우, CloudFront에서 전체 4xx 또는 5xx 상태 코드에 대해 반환하는 값으로 400 또는 500을 지정할 수 있습니다.
- 고객이 웹 사이트가 다운된 것을 모르도록 `200` 상태 코드(OK)와 정적 웹 사이트를 반환하려 할 수 있습니다.

CloudFront 액세스 로그를 활성화하고 응답에서 HTTP 상태 코드를 변경하도록 CloudFront를 구성하는 경우 액세스 로그의 `sc-status` 열 값에는 지정한 상태 코드가 포함됩니다. 하지만 `x-edge-result-type` 열의 값에는 영향이 없습니다. 오리진의 응답 결과 유형이 그대로 포함됩니다. 예를 들어 오리진이 CloudFront에 `404`(찾을 수 없음)를 반환하는 경우 `200`의 코드를 최종 사용자에게 반환하도록 CloudFront를 구성한다고 가정해 보십시오. 오리진이 요청에 `404` 상태 코드로 응답하는 경우 액세스 로그의 `sc-status` 열 값은 `200`이 되지만 `x-edge-result-type` 열의 값은 `Error`가 됩니다.

다음 HTTP 상태 코드와 사용자 지정 오류 페이지를 반환하도록 CloudFront를 구성할 수 있습니다.

- 200
- 400, 403, 404, 405, 414, 416
- 500, 501, 502, 503, 504

CloudFront에서 오류를 캐싱하는 기간 제어

오리진에서 HTTP 4xx 또는 5xx 상태 코드를 반환할 경우, 기본적으로 CloudFront가 이러한 오류 응답을 5분 동안 캐싱합니다. 그런 다음 CloudFront는 객체에 대한 다음 요청을 오리진에 제출하여 오류를 발생시킨 문제가 해결되었는지 여부와 요청된 객체가 현재 사용 가능한지 여부를 확인합니다.

Note

HTTP 상태 코드 416(요청된 범위를 충족할 수 없음)에 대한 사용자 지정 오류 페이지를 만들 수 있으며 오리진에서 CloudFront로 상태 코드 416을 반환할 경우 CloudFront에서 최종 사용자로 반환하는 HTTP 상태 코드를 변경할 수 있습니다. (자세한 내용은 CloudFront에서 반환하는 응답 코드 변경 (p. 257) 단원을 참조하십시오.) 하지만 CloudFront은(는) 상태 코드 416 응답을 캐싱하지 않으며, 따라서 상태 코드 416에 대해 Error Caching Minimum TTL(오류 캐싱 최소 TTL)의 값을 지정할 수 있지만 CloudFront에서는 이 값이 사용되지 않습니다.

CloudFront에서 캐싱하는 4xx 및 5xx 상태 코드 각각에 대해 오류 캐싱 기간(Error Caching Minimum TTL(최소 TTL 캐싱 오류))을 지정할 수 있습니다. 절차는 다음(오류 응답 동작 구성 (p. 259))을 참조하십시오. 기간을 지정할 때는 다음 사항에 유의하십시오.

- 짧은 오류 캐싱 기간을 지정하는 경우, CloudFront에서는 더 긴 기간을 지정하는 경우에 비해 오리진에 더 많은 요청을 전달합니다. 5xx 오류의 경우, 이는 원래 오리진에서 오류를 반환하게 만든 문제를 악화시킬 수 있습니다.
- 오리진에서 객체에 대한 오류를 반환할 경우, CloudFront에서는 오류 캐싱 기간이 경과할 때까지 객체에 대한 요청에 오류 응답이나 사용자 지정 오류 페이지로 응답합니다. 오류 캐싱 기간을 길게 지정한 경우, CloudFront에서는 객체가 다시 사용 가능한 상태가 된 후에도 장시간 오류 응답이나 사용자 지정 오류 페이지로 요청에 계속해서 응답할 수 있습니다.

CloudFront에서 개별 객체에 대한 오류를 캐싱하는 시간을 제어하려는 경우, 다음과 같이 관련 헤더를 해당 객체에 대한 오류 응답에 추가하도록 오리진 서버를 구성할 수 있습니다.

- 오리진에서 **`Cache-Control max-age`** 또는 **`Cache-Control s-maxage`** 명령이나 **`Expires`** 헤더를 추가하는 경우: CloudFront에서는 헤더의 값과 Error Caching Minimum TTL(최소 TTL 캐싱 오류)의 값 중 더 큰 값 동안 오류 응답을 캐싱합니다.

  `Cache-Control max-age` 값과 `Cache-Control s-maxage` 값은 오류 페이지를 가져오는 캐시 동작에 대해 설정된 최대 TTL 값보다 클 수 없습니다.
- 오리진에서 다른 **`Cache-Control`** 명령을 추가하거나 헤더를 추가하지 않는 경우: CloudFront에서는 Error Caching Minimum TTL(최소 TTL 캐싱 오류)의 값 동안 오류 응답을 캐싱합니다.

객체에 대한 4xx 또는 5xx 상태 코드의 만료 시간이 대기하려는 시간보다 더 긴 경우, 요청된 객체의 URL을 사용하여 상태 코드를 무효화할 수 있습니다. 오리진에서 여러 객체에 대한 오류 응답을 반환하려는 경우, 각 객체를 개별적으로 무효화해야 할 수 있습니다. 객체 무효화에 대한 자세한 내용은 파일 무효화 (p. 80) 단원을 참조하십시오.

사용자 지정 오류 페이지를 사용할 수 없을 때 CloudFront에서 응답하는 방식

HTTP 상태 코드에 대해 사용자 지정 오류 페이지를 반환하도록 CloudFront를 구성했으나 사용자 지정 오류 페이지를 사용할 수 없는 경우, CloudFront에서는 사용자 지정 오류 페이지가 포함된 오리진에서 CloudFront로 보낸 상태 코드를 최종 사용자에게 반환합니다. 예를 들어 사용자 지정 오리진에서 500 상태 코드를 반환하고 500 상태 코드에 대한 사용자 지정 오류 페이지를 Amazon S3 버킷에서 가져오도록 CloudFront를 구성했다고 가정합니다. 하지만 누군가 실수로 사용자 지정 오류 페이지를 버킷에서 삭제했고, CloudFront에서는 HTTP 404 상태 코드(찾을 수 없음)를 객체를 요청한 최종 사용자에게 반환합니다.

사용자 지정 오류 페이지에 대한 요금

CloudFront에서 사용자 지정 오류 페이지를 최종 사용자에게 반환하는 경우, 사용자 지정 오류 페이지에 대해 표준 CloudFront 요금을 지불하며 요청된 객체에 대해서는 요금이 부과되지 않습니다. CloudFront 요금에 대한 자세한 내용은 Amazon CloudFront 요금을 참조하십시오.

오류 응답 동작 구성

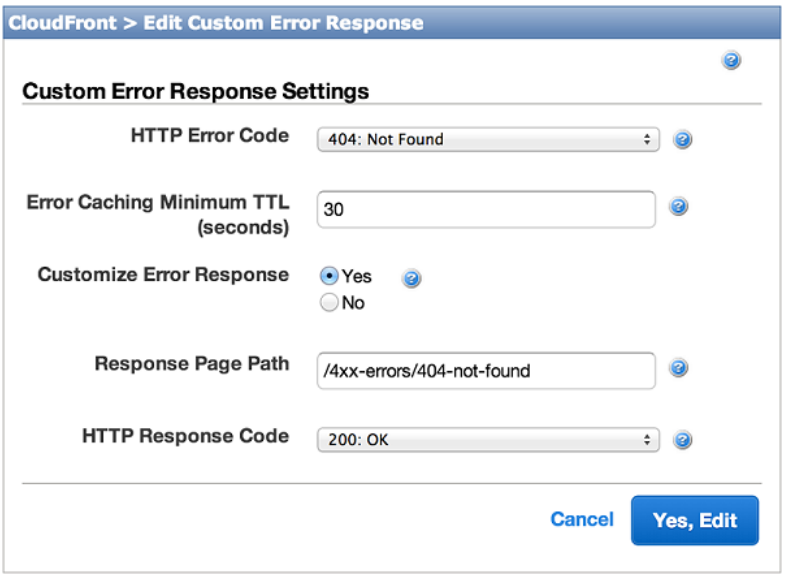
CloudFront API 또는 콘솔을 사용하여 CloudFront 오류 응답을 구성할 수 있습니다. CloudFront API를 사용한 오류 응답 구성에 대한 자세한 내용은 Amazon CloudFront API Reference의 PUT Distribution Config 섹션에서 `CustomErrorResponses` 요소를 참조하십시오.

콘솔을 사용하여 CloudFront 오류 응답을 구성하려면

1. 오리진에서 HTTP 4xx 또는 5xx 오류를 반환할 때 CloudFront에서 최종 사용자에게 반환하려는 사용자 지정 오류 페이지를 만듭니다. CloudFront에서 액세스 가능한 위치에 이 페이지를 저장합니다.

   사용자 지정 오리진을 사용하는 경우라도 사용자 지정 오류 페이지를 Amazon S3 버킷에 저장하는 것이 좋습니다. 사용자 지정 오류 페이지를 HTTP 서버에 저장하는 경우 서버에서 5xx 오류를 반환하기 시작하면 오리진 서버를 사용할 수 없으므로 CloudFront에서는 최종 사용자에게 반환하려는 파일을 가져올 수 없습니다.
2. 최소한 CloudFront에 사용자 지정 오류 페이지 객체에 대한 `read` 권한을 부여했는지 확인하십시오.

   Amazon S3 권한에 대한 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 액세스 제어 단원을 참조하십시오. Amazon S3 콘솔을 사용한 권한 업데이트에 대한 자세한 내용은 Amazon Simple Storage Service 콘솔 사용 설명서를 참조하십시오.
3. (선택 사항) `Cache-Control` 명령 또는 `Expires` 헤더를 특정 객체에 대한 오류 응답과 함께 추가하도록 오리진 서버를 구성합니다(해당되는 경우). 자세한 내용은 CloudFront에서 오류를 캐싱하는 기간 제어 (p. 257) 단원을 참조하십시오.
4. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
5. 배포 목록에서 업데이트하려는 배포를 선택하고 Distribution Settings(배포 설정)를 선택합니다.
6. Error Pages(오류 페이지) 탭을 선택합니다. 그런 다음 사용자 정의 오류 응답 생성를 선택하거나 기존 오류 코드와 편집을 차례로 선택합니다.



7. 관련 값들을 입력합니다. 자세한 내용은 사용자 지정 오류 페이지 및 오류 캐싱 (p. 56) 단원을 참조하십시오.
8. 사용자 지정 오류 페이지를 반환하도록 CloudFront를 구성한 경우, 관련 캐시 동작을 추가하거나 업데이트합니다. 자세한 내용은 다른 위치에 객체 및 사용자 지정 오류 페이지 저장 (p. 257) 단원을 참조하십시오.
9. 변경 사항을 저장하려면 예, 편집합니다를 선택합니다.

CloudFront를 사용한 온디맨드 비디오 및 라이브 스트리밍 비디오

CloudFront를 사용하여 HTTP 오리진에서 온디맨드 비디오(VOD) 또는 라이브 스트리밍 비디오를 제공할 수 있습니다. 클라우드에서 비디오 워크플로를 설정하는 한 가지 방법은 AWS Media Services와 함께 CloudFront를 사용하는 것입니다.

주제

- 스트리밍 비디오: 온디맨드 비디오 및 라이브 스트리밍 (p. 261)
- CloudFront를 사용한 온디맨드 비디오(VOD) 제공 (p. 262)
- CloudFront 및 AWS Media Services를 사용하여 라이브 스트리밍 비디오 제공 (p. 263)
- RTMP 배포 작업 (p. 268)

스트리밍 비디오: 온디맨드 비디오 및 라이브 스트리밍

CloudFront에서 콘텐츠를 배포하려면 먼저 인코더를 사용하여 비디오 콘텐츠를 패키징해야 합니다. 패키징 프로세스에서는 오디오, 비디오 및 캡션 콘텐츠를 포함하는 세그먼트가 생성됩니다. 또한 특정 순서로 재생할 세그먼트와 시점을 설명하는 매니페스트 파일도 생성됩니다. 일반적인 패키지 포맷은 MPEG DASH, Apple HLS, Microsoft Smooth Streaming, CMAF입니다.

온디맨드 비디오(VOD) 스트리밍

온디맨드 비디오(VOD) 스트리밍의 경우, 비디오 콘텐츠가 서버에 저장되고 최종 사용자는 언제든지 비디오 콘텐츠를 볼 수 있습니다. 최종 사용자가 스트리밍할 수 있는 애셋을 만들려면 AWS Elemental MediaConvert와 같은 인코더를 사용하여 미디어 파일을 포맷하고 패키징합니다.

비디오를 올바른 포맷으로 패키징했으면 서버나 Amazon S3 버킷에 저장한 후 최종 사용자들이 요청할 때 CloudFront를 사용하여 제공할 수 있습니다.

라이브 비디오 스트리밍

라이브 비디오 스트리밍의 경우 비디오 콘텐츠가 라이브 이벤트 발생 시 실시간으로 스트리밍되거나 24x7 라이브 채널로 설정됩니다. 브로드캐스트 및 스트리밍 제공을 위한 라이브 출력을 만들려면 AWS Elemental MediaLive와 같은 인코더를 사용하여 비디오를 압축하고 시청 디바이스에 맞게 포맷합니다.

비디오를 인코딩한 후, AWS Elemental MediaStore에 저장하거나 AWS Elemental MediaPackage를 사용하여 다른 전송 형식으로 변환할 수 있습니다. 이러한 오리진을 사용하여 콘텐츠를 제공하도록 CloudFront 배포를 설정합니다. 이러한 서비스와 함께 사용할 배포를 만드는 절차와 지침을 보려면 AWS Elemental MediaStore를 오리진으로 사용하여 비디오 제공 (p. 264) 및 AWS Elemental MediaPackage를 사용하여 포맷된 라이브 비디오 제공 (p. 265) 단원을 참조하십시오.

Wowza 및 Unified Streaming에서도 CloudFront를 통해 비디오를 스트리밍하는 데 사용할 수 있는 도구를 제공합니다. Wowza를 CloudFront와 함께 사용하는 방법은 Wowza 설명서 웹 사이트에서 How to bring your Wowza Streaming Engine license to CloudFront live HTTP streaming(Wowza 스트리밍 엔진 라이선스를 CloudFront 라이브 HTTP 스트리밍으로 가져오는 방법)을 참조하십시오. VOD 스트리밍에 CloudFront와 함께 Unified Streaming을 사용하는 방법에 대한 자세한 내용은 통합 스트리밍 설명서 웹 사이트의 Amazon CloudFront를 참조하십시오.

CloudFront를 사용한 온디맨드 비디오(VOD) 제공

CloudFront를 통해 온디맨드 비디오(VOD) 스트리밍을 제공하려면 다음 서비스를 사용합니다.

- Amazon S3를 사용하여 콘텐츠를 원본 포맷으로 저장하고 트랜스코딩된 비디오 저장.
- 인코더(예: AWS Elemental MediaConvert)를 사용하여 비디오를 스트리밍 포맷으로 트랜스코딩.
- CloudFront를 사용하여 트랜스코딩된 비디오를 최종 사용자에게 제공. Microsoft Smooth Streaming을 사용하려면 Microsoft Smooth Streaming용 온디맨드 비디오 구성 (p. 262) 단원을 참조하십시오.

CloudFront를 사용하여 VOD 솔루션을 만들려면

1. Amazon S3 버킷에 콘텐츠를 업로드합니다. Amazon S3를 사용하는 방법에 대한 자세한 내용은 Amazon Simple Storage Service 개발자 가이드 단원을 참조하십시오.
2. MediaConvert 작업을 사용하여 콘텐츠를 트랜스코딩합니다. 이 작업은 최종 사용자들이 사용할 플레이어에 필요한 포맷으로 비디오를 변환합니다. 작업을 사용하여 다양한 해상도와 비트레이트를 갖는 애셋을 만들 수도 있습니다. 이러한 애셋은 최종 사용자가 사용 가능한 대역폭에 따라 화질을 조정하는 가변 비트레이트(ABR) 스트리밍에 사용됩니다. MediaConvert는 트랜스코딩된 비디오를 S3 버킷에 저장합니다.
3. CloudFront 배포를 사용하여 변환된 콘텐츠를 제공합니다. 최종 사용자는 언제든지 모든 디바이스에서 콘텐츠를 볼 수 있습니다.

   Tip

   AWS CloudFormation 템플릿을 사용하여 VOD AWS 솔루션과 모든 관련 구성 요소를 배포하는 방법을 살펴볼 수 있습니다. 템플릿 사용을 위한 절차를 보려면 AWS 기반 온디맨드 비디오 안내서의 자동 배포를 참조하십시오.

Microsoft Smooth Streaming용 온디맨드 비디오 구성

다음과 같은 방법으로 CloudFront를 사용하여 Microsoft Smooth Streaming 포맷으로 트랜스코딩한 온디맨드 비디오(VOD) 콘텐츠를 배포할 수 있습니다.

- Microsoft IIS를 실행하고 Smooth Streaming을 배포 오리진으로 지원하는 웹 서버를 지정합니다.
- CloudFront 배포의 캐시 동작에서 Smooth Streaming을 활성화합니다. 배포에 여러 캐시 동작을 사용할 수 있기 때문에 하나의 배포를 Smooth Streaming 미디어 파일뿐 아니라 다른 콘텐츠에도 사용할 수 있습니다.

  Important

  Microsoft IIS를 실행하는 웹 서버를 오리진으로 지정하는 경우 CloudFront 배포의 캐시 동작에서 Smooth Streaming을 활성화하지 마십시오. Smooth Streaming을 캐시 동작으로 활성화하면 CloudFront에서 Microsoft IIS 서버를 오리진으로 사용할 수 없습니다.

캐시 동작에서 Smooth Streaming을 활성화할 경우, 즉 Microsoft IIS를 실행하는 서버를 사용하지 않을 경우 다음 사항에 유의하십시오.

- 콘텐츠가 해당 캐시 동작의 경로 패턴 값과 일치하는 경우 동일한 캐시 동작을 사용하여 다른 콘텐츠를 배포할 수 있습니다.
- CloudFront에서는 Smooth Streaming 미디어 파일에 Amazon S3 버킷 또는 사용자 지정 오리진을 사용할 수 있습니다. 캐시 동작에 Smooth Streaming을 활성화하면 CloudFront에서 Microsoft IIS 서버를 사용할 수 없습니다.

- Smooth Streaming 포맷의 미디어 파일은 무효화할 수 없습니다. 만료되기 전에 파일을 업데이트하려는 경우 파일의 이름을 변경해야 합니다. 자세한 내용은 CloudFront가 배포하는 콘텐츠 추가, 제거 또는 바꾸기 (p. 77) 단원을 참조하십시오.

Smooth Streaming 클라이언트에 대한 자세한 내용은 Microsoft 설명서 웹 사이트의 Smooth Streaming Primer를 참조하십시오.

Microsoft IIS 웹 서버가 오리진이 아닌 경우 CloudFront를 사용하여 Smooth Streaming 파일을 배포하려면

1. 미디어 파일을 Smooth Streaming fragmented-MP4 포맷으로 트랜스코딩합니다.
2. 다음 중 하나를 수행하십시오.
   - CloudFront 콘솔을 사용하는 경우: 웹 배포를 생성하거나 업데이트할 때 하나 이상의 배포 캐시 동작에서 Smooth Streaming을 활성화합니다.
   - CloudFront API를 사용하는 경우: 하나 이상의 배포 캐시 동작에 대해 `SmoothStreaming` 요소를 `DistributionConfig` 복합 형식에 추가합니다.
3. Smooth Streaming 파일을 오리진에 업로드합니다.
4. `clientaccesspolicy.xml` 또는 `crossdomainpolicy.xml` 파일을 만들고 이 파일을 배포의 루트에서 액세스할 수 있는 위치(예: `https://d111111abcdef8.cloudfront.net/clientaccesspolicy.xml`)에 추가합니다. 다음은 정책의 예입니다.

```
<?xml version="1.0" encoding="utf-8"?>
<access-policy>
<cross-domain-access>
<policy>
<allow-from http-request-headers="*">
<domain uri="*"/>
</allow-from>
<grant-to>
<resource path="/" include-subpaths="true"/>
</grant-to>
</policy>
</cross-domain-access>
</access-policy>
```

   자세한 내용은 Microsoft Developer Network 웹 사이트의 Making a Service Available Across Domain Boundaries를 참조하십시오.
5. 애플리케이션(예: 미디어 플레이어)의 링크로, 미디어 파일의 URL을 다음 형식으로 지정합니다.

   `https://d111111abcdef8.cloudfront.net/video/presentation.ism/Manifest`

CloudFront 및 AWS Media Services를 사용하여 라이브 스트리밍 비디오 제공

AWS Media Services와 CloudFront를 사용하여 전 세계 시청자에게 라이브 콘텐츠를 제공하려면 이 단원에 나와 있는 지침을 따르십시오.

AWS Elemental MediaLive를 사용하여 라이브 비디오 스트림을 실시간으로 인코딩합니다. 큰 비디오 스트림을 인코딩하는 경우 MediaLive에서 최종 사용자에게 배포할 수 있는 더 작은 버전(인코딩)으로 압축합니다.

라이브 비디오 스트림을 압축한 후 다음 두 가지 주요 방법 중 하나를 사용하여 콘텐츠를 준비하고 제공할 수 있습니다.

- 필요한 포맷으로 콘텐츠를 변환한 후 제공: 여러 포맷의 콘텐츠가 필요한 경우 AWS Elemental MediaPackage를 사용하여 다양한 디바이스 유형에 맞게 콘텐츠를 패키징합니다. 콘텐츠를 패키징할 때 추가 기능을 구현하고 콘텐츠의 무단 사용을 방지하도록 DRM(Digital Rights Management)을 추가할 수도 있습니다. CloudFront를 사용하여 MediaPackage에서 포맷된 콘텐츠를 제공하기 위한 단계별 지침은 AWS Elemental MediaPackage를 사용하여 포맷된 라이브 비디오 제공 (p. 265) 단원을 참조하십시오.
- 확장 가능 오리진을 사용하여 콘텐츠를 저장하고 제공: MediaLive에서 최종 사용자가 사용하는 모든 디바이스에 필요한 포맷으로 콘텐츠를 인코딩한 경우, AWS Elemental MediaStore와 같이 확장성이 우수한 오리진을 사용하여 콘텐츠를 제공할 수 있습니다. MediaStore 컨테이너에 저장된 콘텐츠를 CloudFront를 사용하여 제공하기 위한 단계별 지침은 AWS Elemental MediaStore를 오리진으로 사용하여 비디오 제공 (p. 264) 단원을 참조하십시오.

이러한 옵션 중 하나를 사용하여 오리진을 설정했으면 이제 CloudFront를 사용하여 최종 사용자에게 라이브 스트리밍 비디오를 배포할 수 있습니다.

Tip

가용성이 높은 실시간 시청 환경을 제공하기 위해 자동으로 서비스를 배포하는 AWS 솔루션에 대한 자세한 내용을 알아볼 수 있습니다. 이 솔루션을 자동으로 배포하는 절차를 보려면 라이브 스트리밍 자동 배포 단원을 참조하십시오.

주제

- AWS Elemental MediaStore를 오리진으로 사용하여 비디오 제공 (p. 264)
- AWS Elemental MediaPackage를 사용하여 포맷된 라이브 비디오 제공 (p. 265)

AWS Elemental MediaStore를 오리진으로 사용하여 비디오 제공

AWS Elemental MediaStore 컨테이너에 저장된 비디오가 있으면 CloudFront 배포를 만들어 콘텐츠를 제공할 수 있습니다.

시작하려면 CloudFront에 MediaStore 컨테이너 액세스를 부여합니다. 그리고 CloudFront 배포를 생성하고 MediaStore를 사용하도록 구성합니다.

AWS Elemental MediaStore 컨테이너에서 콘텐츠를 제공하려면

1. Amazon CloudFront의 MediaStore 컨테이너 액세스 허용에 있는 절차를 따르고, 이 단계로 돌아와 배포를 생성합니다.
2. 다음 설정을 사용하여 배포를 생성합니다.

   Origin Domain Name

   MediaStore 컨테이너에 할당된 데이터 엔드포인트. 드롭다운 목록에서 해당 라이브 비디오의 MediaStore 컨테이너를 선택합니다. MediaStore 오리진의 형식은 Container-OriginEndpointURL입니다. 예를 들면 mymediastore.data.mediastore.us-east-1.amazonaws.com과 같습니다. 자세한 내용은 Origin Domain Name (p. 38) 단원을 참조하십시오.

   Origin Path

   객체가 저장되는 MediaStore 컨테이너의 폴더 구조. 자세한 내용은 Origin Path (p. 39) 단원을 참조하십시오.

   오리진 사용자 지정 헤더

   요청을 오리진으로 전달할 때 CloudFront에 사용자 지정 헤더를 포함하고자 하는 경우 헤더 이름과 값을 추가합니다.

최종 사용자 프로토콜 정책

Redirect HTTP to HTTPS(HTTP를 HTTPS로 재지정)를 선택합니다. 자세한 내용은 Viewer Protocol Policy (p. 45) 단원을 참조하십시오.

Object Caching

사용하는 인코더가 모든 객체의 캐시 제어를 설정할 수 없는 경우 Customize(사용자 지정)를 선택합니다. 인코더가 모든 객체의 캐시 제어를 설정할 수 있는 경우 Origin Cache Headers(오리진 캐시 헤더)를 선택합니다.

최소 TTL, 최대 TTL 및 기본 TTL

캐시 수요 및 세그먼트 기간에 따라 적절하게 설정합니다.

Error Caching Minimum TTL

5초 미만으로 설정하면 부실한 콘텐츠 서비스 방지에 도움이 됩니다.

기타 설정의 경우 다른 기술 요구 사항 또는 비즈니스 요구 사항을 기반으로 특정 값을 설정할 수 있습니다. 웹 배포에 대한 모든 옵션 목록과 설정에 대한 내용은 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.

3. CloudFront에서 배포를 프로비저닝한 후 캐시 동작을 편집하여 오리진에 대한 CORS(Cross-Origin Resource Sharing)를 설정합니다.

   1. 배포를 선택한 후 Distribution Settings(배포 설정)를 선택합니다.
   2. 동작을 선택하고, 오리진을 선택한 다음 편집을 선택합니다.
   3. 선택한 요청 헤더 기반의 캐시에서 Whitelist(화이트리스트)를 선택하고, 화이트리스트 헤더에서 오리진을 선택합니다.

   CORS에 대한 자세한 내용은 요청 헤더에 따라 콘텐츠 캐싱 (p. 200)의 CORS(Cross-Origin 리소스 공유) 설정을 준수하도록 CloudFront 구성 단원을 참조하십시오.

4. 미디어 플레이어 등 애플리케이션의 링크에 대해 CloudFront를 사용하여 배포한 다른 객체에 사용한 것과 같은 형식으로 미디어 파일의 이름을 지정합니다.

AWS Elemental MediaPackage를 사용하여 포맷된 라이브 비디오 제공

AWS Elemental MediaPackage를 사용하여 라이브 스트림을 포맷한 경우 CloudFront 배포를 만들고 라이브 스트림을 제공하도록 캐시 동작을 구성하면 됩니다. 다음 절차에서는 이미 MediaPackage를 사용하여 라이브 비디오를 위한 채널을 만들었고 엔드포인트를 추가했다는 전제하에 설명을 진행합니다.

Note

다음 프로세스를 사용하는 대신 MediaPackage에 채널을 저장할 때 CloudFront 배포를 자동으로 생성하도록 선택할 수 있습니다. 자세한 내용은 AWS Elemental MediaPackage 사용 설명서의 AWS Elemental MediaPackage에서 배포 만들기를 참조하십시오.

MediaPackage용 CloudFront 배포를 수동으로 만들려면 다음 단계를 수행합니다.

단계

- 1단계: CloudFront 배포 생성 및 구성 (p. 266)
- 2단계: 다른 엔드포인트를 오리진으로 추가 (p. 266)
- 3단계: 모든 엔드포인트에 대한 캐시 동작 구성 (p. 267)

- 4단계: CloudFront를 사용하여 라이브 스트림 채널 제공 (p. 268)

1단계: CloudFront 배포 생성 및 구성

다음 절차를 완료하여, MediaPackage를 사용하여 생성한 라이브 비디오 채널을 위한 CloudFront 배포를 설정합니다.

라이브 비디오 채널을 위한 배포를 만들려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. [Create Distribution]을 선택합니다.
3. [Select a delivery method for your content]의 [Web] 섹션에서 [Get Started]를 선택합니다.
4. 다음과 같이 배포를 위한 설정을 선택합니다.

   Origin Domain Name

   MediaPackage 라이브 비디오 채널과 엔드포인트가 있는 오리진입니다. 텍스트 필드를 선택한 다음 드롭다운 목록에서 해당 라이브 비디오의 MediaPackage 채널을 선택합니다. 한 개의 채널을 여러 오리진 엔드포인트에 매핑할 수 있습니다.

   다른 AWS 계정을 사용하여 채널을 만든 경우 필드에 해당 오리진 URL 값을 입력하십시오. 오리진은 HTTPS URL이어야 합니다.

   자세한 내용은 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35)의 Origin Domain Name (p. 38) 단원을 참조하십시오.

   Origin Path

   콘텐츠가 제공되는 MediaPackage 엔드포인트의 경로입니다. 오리진 도메인 이름을 선택하면 CloudFront에서 오리진 경로를 채웁니다.

   Origin Domain Name(오리진 도메인 이름)에 다른 AWS 계정의 채널을 사용한 경우에는 Origin Path(오리진 경로) 필드가 자동으로 채워지지 않습니다. 다른 계정에서 올바른 오리진 경로를 확인하여 직접 입력해야 합니다.

   오리진 경로의 작동 방식에 대한 자세한 내용은 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35)의 Origin Path (p. 39) 단원을 참조하십시오.

   그밖의 배포 설정에 대해, 다른 기술 요구 사항 또는 비즈니스 요구 사항을 기반으로 특정 값을 설정합니다. 배포에 대한 모든 옵션 목록과 설정에 대한 내용은 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.

   다른 배포 설정 선택을 마치면 Create Distribution(배포 생성)을 선택합니다.
5. 방금 생성한 배포를 선택한 다음 Behaviors(동작) 탭을 선택합니다.
6. 기본 캐시 동작을 선택하고 오리진에 대해 선택한 채널에 적합한 캐시 동작 설정을 지정합니다. 나중에 한 개 이상의 오리진을 추가한 후 그에 대한 캐시 동작 설정을 편집해 보겠습니다.
7. CloudFront Distributions(CloudFront 배포) 페이지로 이동합니다.
8. 배포의 Status(상태) 열 값이 In Progress(진행 중)에서 Deployed(배포 완료)로 바뀔 때까지 기다립니다. 이렇게 상태가 바뀌면 CloudFront에서 배포를 생성한 것입니다.

2단계: 다른 엔드포인트를 오리진으로 추가

다음 단계를 반복하여 각 MediaPackage 채널 엔드포인트를 배포에 추가합니다.

다른 엔드포인트를 오리진으로 추가하려면

1. CloudFront 콘솔에서 채널에 대해 만든 배포를 선택합니다.
2. Origins and Origin Groups(오리진 및 오리진 그룹) 탭에서 Create Origin(오리진 생성)을 선택합니다.
3. Origin Domain Name(오리진 도메인 이름)의 드롭다운 목록에서 해당 채널의 MediaPackage 엔드포인트를 선택합니다. CloudFront에서 자동으로 Origin Path(오리진 경로) 필드를 채웁니다.
4. 그밖의 설정에 대해, 다른 기술 요구 사항 또는 비즈니스 요구 사항을 기반으로 값을 설정합니다. 자세한 내용은 오리진 설정 (p. 38)의 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.
5. 생성을 선택합니다.

3단계: 모든 엔드포인트에 대한 캐시 동작 구성

각 엔드포인트에 대해 요청을 올바르게 라우팅하는 경로 패턴을 추가하도록 캐시 동작을 구성해야 합니다. 지정하는 경로 패턴은 제공하는 비디오 포맷에 따라 다릅니다. 다음 절차에는 Apple HLS, CMAF, DASH, Microsoft Smooth Streaming 포맷에 대해 사용할 경로 패턴 정보가 나와 있습니다.

일반적으로 각 엔드포인트에 대해 두 개의 캐시 동작을 설정합니다.

- 패턴 매니페스트(해당 파일에 대한 인덱스)
- 세그먼트(비디오 콘텐츠의 파일)

엔드포인트에 대한 캐시 동작을 만들려면

1. CloudFront 콘솔에서 채널에 대해 만든 배포를 선택합니다.
2. Behaviors(동작) 탭에서 Create Behavior(동작 만들기)를 선택합니다.
3. Cache Behavior Settings(캐시 동작 설정) 섹션에서 각 엔드포인트 유형에 대한 다음 지침에 나온 첫 번째 패턴으로 Path Pattern(경로 패턴)을 채웁니다. 예를 들어 DASH 엔드포인트의 경우 Path Pattern(경로 패턴)에 `*.mpd`를 입력합니다.

   경로 패턴

   HLS 엔드포인트의 경우, 다음 두 가지 캐시 동작을 만듭니다.
   - 상위 및 하위 매니페스트에 대해 `*.m3u8`을 사용합니다.
   - 콘텐츠 세그먼트에 대해 `*.ts`를 사용합니다.

   CMAF 엔드포인트의 경우, 다음 두 가지 캐시 동작을 만듭니다.
   - 상위 및 하위 매니페스트에 대해 `*.m3u8`을 사용합니다.
   - 콘텐츠 세그먼트에 대해 `*.mp4`를 사용합니다.

   DASH 엔드포인트의 경우, 다음 두 가지 캐시 동작을 만듭니다.
   - 상위 매니페스트에 대해 `*.mpd`를 사용합니다.
   - 콘텐츠 세그먼트에 대해 `*.mp4`를 사용합니다.

   Microsoft Smooth Streaming 엔드포인트의 경우, 한 개의 매니페스트만 제공되므로 캐시 동작을 `index.ism/*` 하나만 만듭니다.
4. 각 캐시 동작에 대해 다음 설정의 값을 지정합니다.

   Viewer Protocol Policy

   Redirect HTTP to HTTPS(HTTP를 HTTPS로 재지정)를 선택합니다.

   선택한 요청 헤더 기반의 캐시

   None (improves caching)(없음(캐싱 향상))을 선택합니다.

캐싱 향상에 대한 자세한 내용은 CloudFront 엣지 캐시에서 제공되는 요청의 비율 증가(캐시 적중률) (p. 193) 단원을 참조하십시오.

Object Caching

MediaPackage는 정확한 재생 동작을 보장하기 위해 기본 `Cache-Control` 헤더를 설정합니다. 이러한 값을 사용하려면 Use Origin Cache Headers(오리진 캐시 헤더 사용)를 선택합니다. 하지만 비디오 세그먼트의 캐시 시간이 증가할 수 있습니다. 객체가 CloudFront 캐시에 유지되는 시간을 사용자 지정하는 방법은 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35)의 Object Caching (p. 47) 단원을 참조하십시오.

Minimum TTL

5초 미만으로 설정하면 부실한 콘텐츠 서비스 방지에 도움이 됩니다.

Query String Forwarding and Caching

Forward all, cache based on whitelist(모두 전달, 화이트리스트 기반으로 캐싱)을 선택합니다.

Query String Whitelist

CloudFront가 캐싱 기반으로 사용할 쿼리 문자열 파라미터로 `m`을 지정합니다. MediaPackage 응답에는 항상 `?m=###` 태그가 포함되어 엔드포인트의 수정 시간을 캡처합니다. 이 태그에 대해 다른 값을 사용하여 이미 콘텐츠가 캐싱된 경우 CloudFront는 캐싱된 버전을 제공하는 대신 새로운 매니페스트를 요청합니다.

MediaPackage에서 시간 이동 보기 기능을 사용하려는 경우, 매니페스트 요청(`*.m3u8`, `*.mpd` 및 `index.ism/*`)에 대한 캐시 동작에 `start` 및 `end`를 추가 쿼리 문자열 파라미터로 지정합니다. 이렇게 하면 매니페스트 요청에서 요청된 시간 기간에 해당하는 콘텐츠가 제공됩니다. 시간 이동 보기 및 콘텐츠 시작/종료 요청 파라미터 포맷에 대한 자세한 내용은 AWS Elemental MediaPackage 사용 설명서의 시간 이동 보기를 참조하십시오.

5. 생성을 선택합니다.
6. 엔드포인트가 Microsoft Smooth Streaming 엔드포인트가 아닌 경우 Create Behavior(동작 만들기)를 선택한 후 이 단계를 반복하여 두 번째 캐시 동작을 만듭니다.

4단계: CloudFront를 사용하여 라이브 스트림 채널 제공

배포를 만들고 오리진을 추가한 후 캐시 동작을 만들었으면 이제 CloudFront를 사용하여 라이브 스트림 채널을 제공할 수 있습니다. 캐시 동작에 대해 구성한 설정에 따라 CloudFront에서 최종 사용자의 요청을 해당 MediaPackage 엔드포인트로 라우팅합니다.

애플리케이션(예: 미디어 플레이어)의 링크로, CloudFront URL에 독립형 포맷의 미디어 파일의 URL을 지정합니다. 자세한 내용은 CloudFront에서 파일에 대한 URL 형식 사용자 지정 (p. 79) 단원을 참조하십시오.

RTMP 배포 작업

Note

Adobe는 2020년 12월 31일까지 Flash 지원을 종료하기로 결정했습니다. 결과적으로 Amazon CloudFront는 더 이상 Adobe Flash Media Server를 지원하지 않으며 2020년 12월 31일까지 RTMP(Real-Time Messaging Protocol)를 더 이상 배포하지 않을 것입니다. 자세한 내용은 Amazon CloudFront 토론 포럼의 발표 전문을 읽어 보십시오.

이 섹션에서는 RTMP 배포를 구성 및 관리하는 방법을 설명합니다. RTMP 배포는 Adobe Media Server 및 Adobe Real-Time Messaging Protocol(RTMP)을 사용하여 미디어 파일을 스트리밍합니다. RTMP 배포 만들기 방법에 대한 자세한 내용은 RTMP를 사용한 미디어 파일 스트리밍용 작업 목록 (p. 271) 단원을 참조하십시오.

주제

- RTMP 배포 (p. 269)
- RTMP 배포가 작동하는 방식 (p. 269)
- RTMP를 사용한 미디어 파일 스트리밍용 작업 목록 (p. 271)
- CloudFront 콘솔을 사용하여 RTMP 배포 만들기 (p. 271)
- RTMP 배포의 생성 또는 업데이트 시 지정하는 값 (p. 272)
- RTMP 배포의 생성 또는 업데이트 시 CloudFront에서 콘솔에 표시하는 값 (p. 276)
- 미디어 플레이어 구성 (p. 277)
- RTMP 배포에 대해 Amazon S3 버킷을 오리진으로 사용 (p. 278)
- 단일 오리진 서버에 대해 여러 RTMP 배포 생성 (p. 279)
- Crossdomain.xml을 사용한 액세스 제한 (p. 279)
- RTMP 배포 관련 오류 코드 (p. 279)
- RTMP 배포 문제 해결 (p. 280)

RTMP 배포

> **Note**
>
> Adobe는 2020년 12월 31일까지 Flash 지원을 종료하기로 결정했습니다. 결과적으로 Amazon CloudFront는 더 이상 Adobe Flash Media Server를 지원하지 않으며 2020년 12월 31일까지 RTMP(Real-Time Messaging Protocol)를 더 이상 배포하지 않을 것입니다. 자세한 내용은 Amazon CloudFront 토론 포럼의 발표 전문을 읽어 보십시오.

RTMP 배포는 Adobe Media Server 및 Adobe Real-Time Messaging Protocol(RTMP)을 사용하여 미디어 파일을 스트리밍합니다. RTMP 배포에서는 Amazon S3 버킷을 오리진으로 사용해야 합니다.

RTMP 배포 만들기 시 지정하는 값에 대한 자세한 내용은 RTMP 배포 작업 (p. 268) 단원을 참조하십시오. RTMP 배포 만들기에 대한 자세한 내용은 RTMP를 사용한 미디어 파일 스트리밍용 작업 목록 (p. 271) 단원을 참조하십시오.

RTMP 배포가 작동하는 방식

> **Note**
>
> Adobe는 2020년 12월 31일까지 Flash 지원을 종료하기로 결정했습니다. 결과적으로 Amazon CloudFront는 더 이상 Adobe Flash Media Server를 지원하지 않으며 2020년 12월 31일까지 RTMP(Real-Time Messaging Protocol)를 더 이상 배포하지 않을 것입니다. 자세한 내용은 Amazon CloudFront 토론 포럼의 발표 전문을 읽어 보십시오.

CloudFront를 사용하여 미디어 파일을 스트리밍하려면, 다음 두 가지 유형의 파일을 최종 사용자에게 제공합니다.

- 미디어 파일
- JW Player, Flowplayer, Adobe Flash 등의 미디어 플레이어

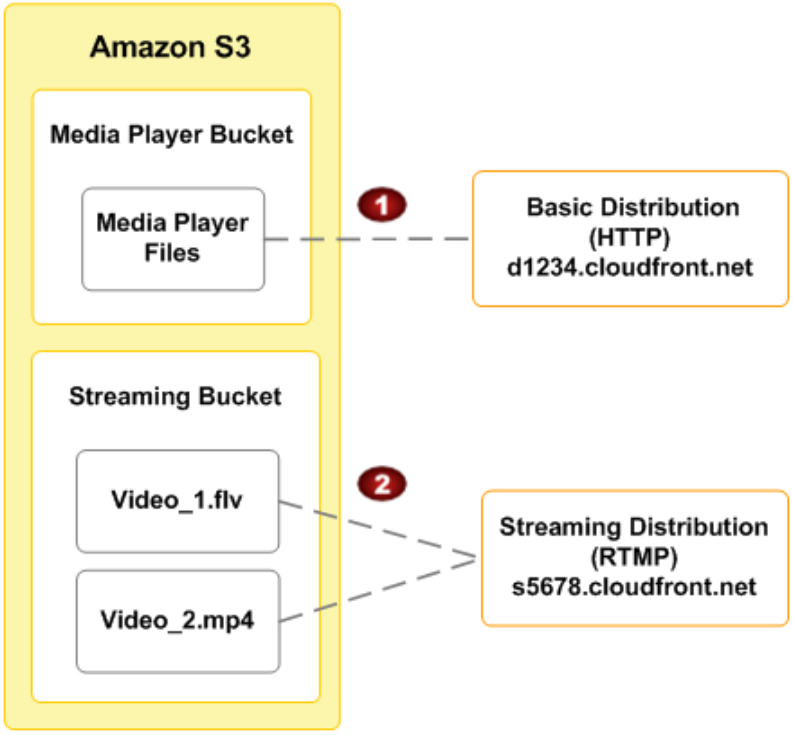
최종 사용자는 해당 미디어 파일용으로 제공한 미디어 플레이어를 사용하여 미디어 파일을 봅니다. 컴퓨터 또는 기타 디바이스에 이미 설치되어 있는 미디어 플레이어(있는 경우)는 사용하지 않습니다.

최종 사용자가 미디어 파일을 스트리밍할 때, 파일이 CloudFront에서 아직 다운로드되고 있는 동안 미디어 플레이어에서 파일의 콘텐츠를 재생하기 시작합니다. 미디어 파일은 최종 사용자의 시스템에 로컬로 저장되지 않습니다.

CloudFront에서 미디어 플레이어와 미디어 파일을 모두 제공하려면, 미디어 플레이어용 웹 배포와 미디어 파일용 RTMP 배포 이렇게 두 가지 유형의 배포가 필요합니다. 웹 배포는 HTTP를 통해 파일을 제공하는 반면, RTMP 배포는 RTMP(또는 RTMP 변형)를 통해 미디어 파일을 스트리밍합니다.

다음 예에서는 미디어 파일과 미디어 플레이어가 Amazon S3의 서로 다른 버킷에 저장되어 있다고 가정합니다만, 이는 필수 사항은 아닙니다. 미디어 파일과 미디어 플레이어를 동일 Amazon S3 버킷에 저장할 수 있습니다. 또한 미디어 플레이어를 최종 사용자가 다른 방식(예: CloudFront 및 사용자 지정 오리진을 사용하여)으로 사용할 수 있도록 할 수 있습니다. 그러나 미디어 파일은 Amazon S3 버킷을 오리진으로 사용해야 합니다.

다음 다이어그램에서 사이트는 미디어 플레이어의 캐싱된 사본을 `d1234.cloudfront.net` 도메인을 통해 각 최종 사용자에게 제공합니다. 그런 뒤 미디어 플레이어는 `s5678.cloudfront.net` 도메인을 통해 미디어 파일의 캐싱된 사본에 액세스합니다.



1. 미디어 플레이어 버킷은 미디어 플레이어를 보관하며 일반적인 HTTP 배포용 오리진 서버가 됩니다. 이 예에서 배포에 대한 도메인 이름은 `d1234.cloudfront.net`입니다. (d의 `d1234.cloudfront.net`는 이것이 웹 배포임을 나타냅니다.)
2. 스트리밍 미디어 버킷은 미디어 파일을 보관하며 RTMP 배포용 오리진 서버가 됩니다. 이 예에서 배포에 대한 도메인 이름은 `s5678.cloudfront.net`입니다. (s의 `s5678.cloudfront.net`는 이것이 RTMP 배포임을 나타냅니다.)

미디어 파일을 배포하도록 CloudFront를 구성할 경우, CloudFront에서는 Adobe Flash Media Server 스트리밍 서버로 사용하여 Adobe RTMP(실시간 메시징 프로토콜)를 통해 미디어 파일을 스트리밍합니다. CloudFront에서는 포트 1935 및 포트 80을 통해 RTMP 요청을 수락합니다.

CloudFront는 다음 RTMP 프로토콜 변형을 지원합니다.

- RTMP – Adobe 실시간 메시징 프로토콜
- RTMPT – HTTP를 통해 터널링된 Adobe 스트리밍
- RTMPE – Adobe 암호화
- RTMPTE – HTTP를 통해 터널링된 Adobe 암호화

RTMP를 사용한 미디어 파일 스트리밍용 작업 목록

> **Note**
>
> Adobe는 2020년 12월 31일까지 Flash 지원을 종료하기로 결정했습니다. 결과적으로 Amazon CloudFront는 더 이상 Adobe Flash Media Server를 지원하지 않으며 2020년 12월 31일까지 RTMP(Real-Time Messaging Protocol)를 더 이상 배포하지 않을 것입니다. 자세한 내용은 Amazon CloudFront 토론 포럼의 발표 전문을 읽어 보십시오.

다음 작업 목록에는 미디어 플레이어용 Adobe RTMP 프로토콜을 사용하여 온디맨드 비디오(VOD) 스트리밍 배포를 만드는 프로세스가 요약되어 있습니다.

RTMP 배포를 생성하려면

1. 미디어 파일용 Amazon S3 버킷을 만듭니다. 미디어 플레이어용으로 다른 Amazon S3 버킷을 사용하는 경우, 미디어 플레이어 파일용 Amazon S3 버킷도 하나 만듭니다.

   버킷의 이름은 전부 소문자로만 구성되어야 하며 공백을 포함할 수 없습니다.
2. 미디어 파일을 재생할 미디어 플레이어를 선택 및 구성합니다. 자세한 내용은 해당 미디어 플레이어의 설명서를 참조하십시오.
3. 미디어 플레이어용 파일을 CloudFront에서 파일을 가져오게 하려는 오리진에 업로드합니다. 미디어 플레이어용 오리진으로 Amazon S3 버킷을 사용하는 경우, 버킷이 아닌 파일을 공개적으로 읽기 가능하도록 설정합니다.
4. 미디어 플레이어용 웹 배포를 만듭니다. (기존 배포를 사용해도 됩니다.) 자세한 내용은 배포 만들기 단계(개요) (p. 34) 단원을 참조하십시오.
5. 미디어 파일용으로 만든 Amazon S3 버킷에 미디어 파일을 업로드하고, 버킷이 아닌 콘텐츠를 공개적으로 읽기 가능하도록 설정합니다.

   > **Important**
   >
   > Flash Video 컨테이너에 있는 미디어 파일은 .flv 파일 이름 확장자를 포함해야 하며, 그렇지 않으면 미디어가 스트리밍되지 않습니다.

   미디어 플레이어 파일과 미디어 파일을 동일 버킷에 넣을 수 있습니다.
6. 미디어 파일용 RTMP 배포를 만듭니다.

   - CloudFront 콘솔을 사용하여 RTMP 배포를 만드는 방법에 대한 자세한 내용은 CloudFront 콘솔을 사용하여 RTMP 배포 만들기 (p. 271) 단원을 참조하십시오.
   - CloudFront API를 사용하여 RTMP 배포를 만드는 방법에 대한 자세한 내용은 Amazon CloudFront API Reference의 CreateStreamingDistribution을 참조하십시오.
7. 미디어 플레이어를 구성합니다. 자세한 내용은 미디어 플레이어 구성 (p. 277) 단원을 참조하십시오.

콘텐츠를 재생하는 데에 문제가 있는 경우 RTMP 배포 문제 해결 (p. 280) 단원을 참조하십시오.

CloudFront 콘솔을 사용하여 RTMP 배포 만들기

> **Note**
>
> Adobe는 2020년 12월 31일까지 Flash 지원을 종료하기로 결정했습니다. 결과적으로 Amazon CloudFront는 더 이상 Adobe Flash Media Server를 지원하지 않으며 2020년 12월 31일까지 RTMP(Real-Time Messaging Protocol)를 더 이상 배포하지 않을 것입니다. 자세한 내용은 Amazon CloudFront 토론 포럼의 발표 전문을 읽어 보십시오.

다음 절차는 CloudFront 콘솔을 사용하여 RTMP 배포를 만드는 방법을 설명합니다. CloudFront API를 사용하여 RTMP 배포를 만들려는 경우 Amazon CloudFront API Reference에서 CreateStreamingDistribution을 참조하십시오.

각 AWS 계정에 대해 생성할 수 있는 RTMP 배포 수에 대한 현재 할당량(이전에는 제한이라고 함)은 할당량 (p. 447) 단원을 참조하십시오. 더 높은 할당량을 요청하려면 https://console.aws.amazon.com/support/home#/case/create?issueType=service-limit-increase&limitType=service-code-cloudfront-distributions로 이동하십시오.

CloudFront 콘솔을 사용하여 RTMP 배포를 생성하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. Create Distribution(배포 생성)을 클릭합니다.
3. Create Distribution Wizard(배포 생성 마법사)의 첫 번째 페이지에서 RTMP 섹션의 시작하기를 선택합니다.
4. 배포에 대해 설정을 지정합니다. 자세한 내용은 RTMP 배포의 생성 또는 업데이트 시 지정하는 값 (p. 272) 단원을 참조하십시오.
5. Create Distribution(배포 생성)을 클릭합니다.
6. CloudFront에서 배포를 생성하면 배포에 대한 상태 열의 값이 진행 중에서 배포 완료로 변경됩니다. 배포를 활성화하도록 선택한 후에는 요청을 처리할 수 있습니다. 이때 걸리는 시간은 15분 미만입니다.

   CloudFront가 배포에 할당하는 도메인 이름이 배포 목록에 나타납니다. 이 도메인 이름은 선택한 배포에 대한 일반 탭에도 나타납니다.

RTMP 배포의 생성 또는 업데이트 시 지정하는 값

> **Note**
>
> Adobe는 2020년 12월 31일까지 Flash 지원을 종료하기로 결정했습니다. 결과적으로 Amazon CloudFront는 더 이상 Adobe Flash Media Server를 지원하지 않으며 2020년 12월 31일까지 RTMP(Real-Time Messaging Protocol)를 더 이상 배포하지 않을 것입니다. 자세한 내용은 Amazon CloudFront 토론 포럼의 발표 전문을 읽어 보십시오.

CloudFront를 사용하여 미디어 파일을 스트리밍하려면, RTMP 배포를 만들어 다음 값을 지정합니다.

주제

- Origin Domain Name(Amazon S3 버킷) (p. 273)
- Restrict Bucket Access(Amazon S3에만 해당) (p. 273)
- Origin Access Identity(Amazon S3에만 해당) (p. 273)
- Comment for New Identity(Amazon S3에만 해당) (p. 273)
- Your Identities(Amazon S3에만 해당) (p. 273)
- Grant Read Permissions on Bucket(Amazon S3에만 해당) (p. 274)
- Price Class (p. 274)
- Alternate Domain Names (CNAMEs) (p. 274)
- Logging (p. 274)
- Bucket for Logs (p. 274)
- Log Prefix (p. 275)
- Comment (p. 275)
- Distribution State (p. 275)
- Restrict Viewer Access (Use Signed URLs) (p. 275)
- Trusted Signers (p. 275)
- AWS Account Numbers (p. 276)

Origin Domain Name(Amazon S3 버킷)

CloudFront에서 이 오리진에 대한 객체를 가져오게 하려는 Amazon S3 버킷의 DNS 도메인 이름입니다(예: `aws-s3-bucket1.s3.amazonaws.com`). CloudFront 콘솔에서 Origin Domain Name(오리진 도메인 이름) 필드를 클릭하면 현재 AWS 계정과 연결된 Amazon S3 버킷을 열거하는 목록이 나타납니다. 다른 AWS 계정에서 버킷을 사용하려면 다음 형식으로 버킷의 도메인 이름을 입력합니다.

`bucket-name.s3.region.amazonaws.com`

버킷에 Amazon S3 Transfer Acceleration을 구성한 경우, Origin Domain Name(오리진 도메인 이름)에 `s3-accelerate` 엔드포인트를 지정하지 마십시오.

CloudFront 원본 액세스 ID를 사용하여 Amazon S3의 콘텐츠에 보안을 적용하지 않는 한, 파일은 공개적으로 읽기 가능해야 합니다. 자세한 내용은 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오.

> **Important**
>
> 버킷 이름은 DNS 명명 요구 사항을 준수하는 이름을 사용해야 합니다. 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 버킷 규제 및 제한을 참조하십시오.

CloudFront에서 현재 오리진용 객체를 가져오는 버킷을 변경할 경우, CloudFront에서는 CloudFront 엣지 로케이션에 대한 변경 사항 복제를 즉시 시작합니다. 배포 구성이 지정된 엣지 로케이션에서 업데이트될 때까지 CloudFront에서는 계속해서 요청을 이전 Amazon S3 버킷으로 전달합니다. 배포 구성이 해당 엣지 로케이션에서 업데이트되는 즉시 CloudFront에서는 요청을 새 Amazon S3 버킷으로 전달하기 시작합니다.

버킷을 변경하더라도 CloudFront에서는 엣지 캐시를 새 오리진의 객체로 다시 채울 필요가 없습니다. 애플리케이션의 최종 사용자 요청이 변경되지 않는 한, CloudFront에서는 각 객체의 TTL이 만료되거나 자주 요청되지 않는 객체가 제거될 때까지는 계속해서 엣지 캐시에 이미 존재하는 객체를 제공합니다.

자세한 내용은 RTMP 배포에 대해 Amazon S3 버킷을 오리진으로 사용 (p. 278) 단원을 참조하십시오.

Restrict Bucket Access(Amazon S3에만 해당)

최종 사용자에게 Amazon S3 URL이 아닌 CloudFront URL만 사용하여 Amazon S3 버킷의 객체에 액세스하게 하려면 예를 클릭합니다. 그런 다음 관련 값들을 지정합니다.

최종 사용자가 CloudFront URL 또는 Amazon S3 URL을 사용하여 객체에 액세스할 수 있도록 하려면 아니요를 클릭합니다.

자세한 내용은 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오.

Origin Access Identity(Amazon S3에만 해당)

Restrict Bucket Access(버킷 액세스 제한)에 대해 예를 선택한 경우, 새로운 원본 액세스 ID를 만들지 아니면 AWS 계정에 연결된 기존 원본 액세스 ID를 사용할지를 선택합니다. 원본 액세스 ID가 이미 있는 경우 이것을 재사용하여 유지 보수를 단순화하는 것이 좋습니다. 원본 액세스 ID에 대한 자세한 내용은 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오.

Comment for New Identity(Amazon S3에만 해당)

Origin Access Identity(원본 액세스 ID)에 대해 Create a New Identity(새 자격 증명 생성)를 선택한 경우, 새 원본 액세스 ID를 식별하는 설명을 입력합니다. 이 배포를 생성할 때 CloudFront에서는 해당하는 오리진 액세스 자격 증명을 생성합니다.

Your Identities(Amazon S3에만 해당)

Origin Access Identity(원본 액세스 ID)에 대해 Use an Existing Identity(기존 자격 증명 사용)를 선택한 경우, 사용하려는 원본 액세스 ID를 선택합니다. 다른 AWS 계정과 연결된 원본 액세스 ID는 사용할 수 없습니다.

Grant Read Permissions on Bucket(Amazon S3에만 해당)

CloudFront에서 원본 액세스 ID에 자동으로 Amazon S3 버킷에 있는 객체에 대한 읽기 권한을 부여하려는 경우, Yes, Update Bucket Policy(예, 버킷 정책을 업데이트합니다)를 클릭합니다.

> **Important**
>
> Yes, Update Bucket Policy(예, 버킷 정책을 업데이트합니다)를 클릭한 경우, CloudFront에서는 지정한 원본 액세스 ID에 버킷의 객체에 대한 읽기 권한을 부여하도록 버킷 정책을 업데이트합니다. 그러나, CloudFront에서는 버킷 정책에 있는 기존 권한이나 개별 객체에 대한 권한을 제거하지는 않습니다. 즉, 사용자가 Amazon S3 URL을 사용하는 버킷의 객체에 대해 현재 액세스 권한이 있는 경우, CloudFront에서 버킷 정책을 업데이트한 후에도 해당 권한은 유지됩니다. 기존 버킷 정책과 버킷의 객체에 대한 기존 권한을 확인 또는 변경하려면 Amazon S3에서 제공되는 메소드를 사용하십시오. 자세한 내용은 OAI에 Amazon S3 버킷 내 파일 읽기 권한 부여 (p. 179) 단원을 참조하십시오.

버킷 권한을 업데이트하는 대신 객체의 ACL을 업데이트하려는 경우와 같이 권한을 수동으로 업데이트하려는 경우, No, I will Update Permissions(아니요, 권한을 업데이트하겠습니다)를 클릭합니다.

Price Class

CloudFront 서비스를 위해 지불하려는 최고 가격에 해당하는 가격 등급입니다. 기본적으로 CloudFront는 모든 CloudFront 지역의 에지에서 객체를 제공합니다.

가격 등급과 가격 등급 선택이 배포를 위한 CloudFront 성능에 미치는 영향에 대한 자세한 내용은 CloudFront 배포의 가격 등급 선택 (p. 9) 단원을 참조하십시오. 가격 등급이 CloudFront 지역에 매핑되는 방식을 포함해 CloudFront 요금에 대한 자세한 내용은 Amazon CloudFront 요금을 참조하십시오.

Alternate Domain Names (CNAMEs)

선택. 배포를 만들 때 CloudFront에서 할당한 도메인 이름을 사용하는 대신 URL의 도메인 이름(예: example.com)을 객체에 사용할 수 있도록 하나 이상의 CNAME 별칭을 배포와 연결할 수 있습니다. 자세한 내용은 대체 도메인 이름(CNAME)을 추가하여 파일에 대해 사용자 지정 URL 사용 (p. 66) 단원을 참조하십시오.

Logging

CloudFront에서 객체에 대한 각 요청의 정보를 로깅하고 이 로그 파일을 Amazon S3 버킷에 저장할지 여부입니다. 언제든지 로깅을 활성화 또는 비활성화할 수 있습니다. 로깅을 활성화해도 추가 비용이 들지는 않지만 파일을 Amazon S3 버킷에 저장하고 액세스하는 데 일반 Amazon S3 요금이 발생합니다. 로그는 언제든지 삭제할 수 있습니다. CloudFront 액세스 로그에 대한 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

Bucket for Logs

로깅에 대해 설정을 선택한 경우, CloudFront에서 액세스 로그를 저장하려는 대상 Amazon S3 버킷입니다(예: `myawslogbucket.s3.amazonaws.com`). 로깅을 활성화한 경우, CloudFront에서는 객체에 대한 각 최종 사용자 요청의 정보를 기록하고 이 파일을 지정된 Amazon S3 버킷에 저장합니다. 언제든지 로깅을 활성화 또는 비활성화할 수 있습니다. CloudFront 액세스 로그에 대한 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

> **Note**
>
> Amazon S3 버킷 ACL을 가져오고 업데이트할 수 있는 허가가 필요하며, 해당 버킷의 S3 ACL에서 사용자에게 `FULL_CONTROL`을 부여해야 합니다. 이렇게 하면 CloudFront에서 버킷에 로그 파일을 저장할 수 있는 awsdatafeeds 계정 허가를 부여합니다. 자세한 내용은 로그 기록 및 로그 파일에 대한 액세스를 구성하는 데 필요한 권한 (p. 395) 단원을 참조하십시오.

Log Prefix

선택. 로깅에 대해 설정을 선택한 경우, CloudFront에서 이 배포에 대한 액세스 로그 파일 이름에 접두사로 지정하려는 문자열(있는 경우)을 지정합니다(예: `exampleprefix/`). 후행 슬래시(/)는 선택 사항이지만 로그 파일을 쉽게 검색할 수 있도록 넣는 것이 좋습니다. CloudFront 액세스 로그에 대한 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

Comment

선택. 배포를 만들 때 최대 128자의 설명을 포함할 수 있습니다. 설명은 언제든지 업데이트할 수 있습니다.

Distribution State

배포를 만들 때 배포가 생성된 후에 이 배포를 활성화할지 또는 비활성화할지 여부를 지정해야 합니다.

- 활성은 배포가 완료되는 대로 이 배포의 도메인 이름을 사용하는 링크를 배포할 수 있고 최종 사용자가 콘텐츠를 가져올 수 있음을 의미합니다. 배포가 활성화되어 있으면 CloudFront에서는 해당 배포와 연결된 도메인 이름을 사용하는 콘텐츠에 대해 최종 사용자 요청을 수락하고 처리합니다.

  CloudFront 배포를 생성, 수정 또는 삭제할 경우 CloudFront 데이터베이스에 변경 사항이 전파되는 데에는 다소 시간이 소요됩니다. 배포 정보에 대한 즉각적인 요청에는 변경 사항이 반영되지 않을 수 있습니다. 전파는 대개 몇 분 내에 완료되지만, 이때 시스템에 과도한 로드가 걸리거나 네트워크 파티션이 증가할 수 있습니다.
- 비활성화는 배포가 완료되어 사용할 준비가 되더라도 최종 사용자가 이를 사용할 수 없음을 의미합니다. 배포가 비활성화되어 있으면 CloudFront에서는 해당 배포와 연결된 도메인 이름을 사용하는 최종 사용자 요청을 수락하지 않습니다. 배포의 구성을 업데이트하여 비활성 상태에서 활성 상태로 전환할 때까지는 누구도 이 배포를 사용할 수 없습니다.

원하는 만큼 배포를 비활성 상태와 활성 상태 간에 전환할 수 있습니다. 배포 구성 업데이트에 대한 자세한 내용은 배포 업데이트 (p. 59) 단원을 참조하십시오.

Restrict Viewer Access (Use Signed URLs)

이 배포로 제공되는 객체에 대한 요청에 퍼블릭 URL을 사용하길 원하는 경우 아니요를 클릭합니다. 요청에 서명된 URL을 사용하려는 경우 예를 클릭합니다. 그런 다음 서명된 URL 생성에 사용하려는 AWS 계정을 지정합니다. 이러한 계정은 신뢰할 수 있는 서명자라고도 합니다.

신뢰할 수 있는 서명자에 대한 자세한 내용은 서명된 URL 및 서명된 쿠키(신뢰할 수 있는 서명자)를 만들 수 있는 AWS 계정 지정 (p. 118) 단원을 참조하십시오.

Trusted Signers

이 배포에 대해 신뢰할 수 있는 서명자로 사용하려는 AWS 계정을 선택합니다.

- 셀프: 현재 AWS Management 콘솔에 신뢰할 수 있는 서명자로 로그인된 계정을 사용합니다. 현재 IAM 사용자로 로그인되어 있는 경우, 연결된 AWS 계정이 신뢰할 수 있는 서명자로 추가됩니다.
- Specify Accounts(계정 지정): 신뢰할 수 있는 서명자에 대한 계정 번호를 AWS Account Numbers(AWS 계정 번호) 필드에 입력합니다.

서명된 URL을 생성하려면 AWS 계정에 하나 이상의 활성 CloudFront 키 페어가 있어야 합니다.

> **Important**
>
> 콘텐츠 배포에 이미 사용 중인 배포를 업데이트하려는 경우, 서명된 URL 생성을 시작할 준비가 되었을 때만 신뢰할 수 있는 서명자를 추가합니다. 배포에 신뢰할 수 있는 서명자를 추가한 후에, 사용자는 서명된 URL을 사용하여 이 배포로 제공되는 객체에 액세스해야 합니다.

AWS Account Numbers

현재 계정에 추가로 또는 현재 계정 대신에 AWS 계정을 사용하여 서명된 URL을 생성하려는 경우, 이 필드에 한 줄당 AWS 계정 번호 하나씩을 입력합니다. 다음을 참조하십시오.

- 지정하는 계정에는 하나 이상의 활성 CloudFront 키 페어가 있어야 합니다. 자세한 내용은 신뢰할 수 있는 서명자에 대해 CloudFront 키 페어 생성 (p. 119) 단원을 참조하십시오.
- IAM 사용자에 대해 CloudFront 키 페어를 생성할 수 없으므로 IAM 사용자는 신뢰할 수 있는 서명자로 사용할 수 없습니다.
- 계정에 대해 AWS 계정 번호를 가져오는 방법에 대한 자세한 내용은 Amazon Web Services 일반 참조의 보안 자격 증명을 가져오는 방법 단원을 참조하십시오.
- 현재 계정에 대한 계정 번호를 입력하면 CloudFront에서는 셀프 확인란이 자동으로 선택되고 AWS Account Numbers(AWS 계정 번호) 목록에서 계정 번호가 삭제됩니다.

RTMP 배포의 생성 또는 업데이트 시 CloudFront에서 콘솔에 표시하는 값

> **Note**
>
> Adobe는 2020년 12월 31일까지 Flash 지원을 종료하기로 결정했습니다. 결과적으로 Amazon CloudFront는 더 이상 Adobe Flash Media Server를 지원하지 않으며 2020년 12월 31일까지 RTMP(Real-Time Messaging Protocol)를 더 이상 배포하지 않을 것입니다. 자세한 내용은 Amazon CloudFront 토론 포럼의 발표 전문을 읽어 보십시오.

새 RTMP 배포를 생성하거나 기존 배포를 업데이트할 때, CloudFront에서는 다음 정보를 CloudFront 콘솔에 표시합니다.

> **Note**
>
> 활성화된 신뢰할 수 있는 서명자, 활성 CloudFront 키 페어를 보유하고 있으며 유효한 서명된 URL을 만드는 데 사용할 수 있는 AWS 계정은 현재 CloudFront 콘솔에 표시되지 않습니다.

배포 ID

CloudFront API를 사용하여 배포에 대한 작업을 수행할 경우, 배포 ID를 사용하여 작업을 수행하려는 대상 배포를 지정합니다(예: `EDFDVBD6EXAMPLE`). 배포 ID는 변경할 수 없습니다.

상태

배포에 대해 가능한 상태 값은 다음 표에 나와 있습니다.

값	설명
InProgress	배포가 아직 생성되거나 업데이트되는 중입니다.
배포 완료	배포의 생성 또는 업데이트가 완료되어 변경 사항이 CloudFront 시스템 전반에 완전히 전파되었습니다.

배포의 상태가 배포 완료가 되어야 하는 것에 추가로, 최종 사용자가 CloudFront를 사용하여 콘텐츠에 액세스할 수 있으려면 먼저 배포를 활성화해야 합니다. 자세한 내용은 Distribution State (p. 275) 단원을 참조하십시오.

마지막 수정

배포가 마지막으로 수정된 날짜 및 시간을 ISO 8601 형식을 사용하여 나타냅니다(예: 2012-05-19T19:37:58Z). 자세한 내용은 http://www.w3.org/TR/NOTE-datetime을 참조하십시오.

도메인 이름

객체에 대한 링크에 배포의 도메인 이름을 사용합니다. 그렇지 않은 경우 대체 도메인 이름(CNAME)을 사용합니다. 예를 들어 배포의 도메인 이름이 `d111111abcdef8.cloudfront.net`인 경우, 예제 `/images/image.jpg` 파일에 대한 링크는 `http://d111111abcdef8.cloudfront.net/images/image.jpg`가 됩니다. 배포에 대한 CloudFront 도메인 이름은 변경할 수 없습니다. 객체에 대한 링크의 CloudFront URL에 대한 자세한 내용은 CloudFront에서 파일에 대한 URL 형식 사용자 지정 (p. 79) 단원을 참조하십시오.

하나 이상의 대체 도메인 이름(CNAME)을 지정한 경우, 객체에 대한 링크에 CloudFront 도메인 이름을 사용하는 대신 자체 도메인 이름을 사용할 수 있습니다. CNAME에 대한 자세한 내용은 Alternate Domain Names (CNAMEs) (p. 51)을 참조하십시오.

> **Note**
>
> CloudFront 도메인 이름은 고유합니다. 즉, 배포의 도메인 이름은 이전 배포에 사용된 적이 없고 이후 다른 배포에 재사용되지도 않습니다.

미디어 플레이어 구성

> **Note**
>
> Adobe는 2020년 12월 31일까지 Flash 지원을 종료하기로 결정했습니다. 결과적으로 Amazon CloudFront는 더 이상 Adobe Flash Media Server를 지원하지 않으며 2020년 12월 31일까지 RTMP(Real-Time Messaging Protocol)를 더 이상 배포하지 않을 것입니다. 자세한 내용은 Amazon CloudFront 토론 포럼의 발표 전문을 읽어 보십시오.

미디어 파일을 재생하려면 파일에 대한 정확한 경로와 함께 미디어 플레이어를 구성해야 합니다. 미디어를 구성하는 방법은 사용하려는 미디어 플레이어와 이를 사용할 방법에 따라 달라집니다.

미디어 플레이어를 구성할 때 미디어 파일에 지정하는 경로에는 다음과 같이 도메인 이름 바로 다음에 `cfx/st` 문자가 포함되어야 합니다.

`rtmp://s5c39gqb8ow64r.cloudfront.net/cfx/st/mediafile.flv`를 선택하십시오.

> **Note**
>
> CloudFront에서는 Adobe의 FMS 명명 요구 사항을 따릅니다. 플레이어별로 스트림을 지정하는 방법에 관한 자체 규칙을 갖고 있습니다. 상기 예는 JW Player용입니다. 해당 플레이어의 설명서를 확인하세요. 예를 들어, Adobe Flash Media Server에서는 재생 경로에 `.flv` 확장명을 표시할 수 없습니다. 대부분의 플레이어에서는 `.flv` 확장명이 자동으로 제거됩니다.

미디어 플레이어에서 파일 이름과 별도로 경로를 요청할 수 있습니다. 예를 들어, JW Player를 통해 `streamer` 및 `file` 변수를 다음과 같이 지정합니다.

- streamer – `rtmp://s5c39gqb8ow64r.cloudfront.net/cfx/st` (후행 슬래시 없음)
- 파일 – `mediafile.flv`

미디어 파일을 버킷 내 디렉터리 저장한 경우(예: `videos/mediafile.flv`), JW Player용 변수는 다음과 같습니다.

- streamer – `rtmp://s5c39gqb8ow64r.cloudfront.net/cfx/st` (후행 슬래시 없음)

- 파일 – `videos/mediafile.flv`

JW Player에 대한 자세한 내용은 JW Player 웹 사이트를 참조하십시오.

MPEG 파일

MP3 오디오 파일 또는 H.264/MPEG-4 비디오 파일을 제공하려면 파일 이름에 `mp3:` 또는 `mp4:` 접두사를 적용해야 할 수 있습니다. 일부 미디어 플레이어에서는 자동으로 해당 접두사가 추가되도록 구성되어 있습니다. 그러한 미디어 플레이어에서는 또한 파일 확장명 없이(예: `magicvideo` 대신 `magicvideo.mp4`) 파일 이름을 지정하도록 요구할 수 있습니다.

RTMP 배포에 대해 Amazon S3 버킷을 오리진으로 사용

> **Note**
>
> Adobe는 2020년 12월 31일까지 Flash 지원을 종료하기로 결정했습니다. 결과적으로 Amazon CloudFront는 더 이상 Adobe Flash Media Server를 지원하지 않으며 2020년 12월 31일까지 RTMP(Real-Time Messaging Protocol)를 더 이상 배포하지 않을 것입니다. 자세한 내용은 Amazon CloudFront 토론 포럼의 발표 전문을 읽어 보십시오.

배포를 생성할 때 CloudFront에서 엣지 로케이션에 배포할 파일을 가져오는 위치를 지정합니다. RTMP 배포의 경우 Amazon S3 버킷을 사용해야 합니다. 사용자 지정 오리진은 지원되지 않습니다. 버킷으로 객체를 가져오기 위해 Amazon S3 API 또는 타사 도구 등 Amazon S3에서 지원되는 모든 메서드를 사용할 수 있습니다. 다른 Amazon S3 버킷과 마찬가지로 버킷에 계층을 만들 수 있습니다. 버킷에 객체를 저장하기 위해서는 정규 Amazon S3 요금이 발생합니다. CloudFront 사용을 위한 요금에 대한 자세한 내용은 콘솔의 CloudFront 보고서 (p. 369) 단원을 참조하십시오.

기존 Amazon S3 버킷을 CloudFront 오리진 서버로 사용하더라도 버킷은 어떠한 방식으로도 변경되지 않습니다. 평상시처럼 Amazon S3 객체를 저장 및 액세스할 용도로 보통 Amazon S3 가격에 버킷을 계속해서 사용할 수 있습니다.

RTMP 배포와 웹 배포 모두에 동일한 Amazon S3 버킷을 사용할 수 있습니다.

> **Note**
>
> RTMP 배포를 만든 후에는 이 배포의 오리진 서버를 변경할 수 없습니다. RTMP 배포용 Amazon S3 버킷을 변경하려는 경우, 새 버킷을 사용하는 새 배포를 만들고 새 배포에 이 도메인 이름을 사용하도록 링크 또는 DNS 레코드를 업데이트해야 합니다. 그런 다음 원래 배포를 삭제할 수 있습니다. 자세한 내용은 배포 삭제 (p. 60) 단원을 참조하십시오.

CloudFront가 객체를 가져오고자 하는 Amazon S3 버킷을 지정할 때 다음 형식을 사용하여 버킷에 액세스하는 것이 좋습니다.

```
bucket-name.s3.region.amazonaws.com
```

다른 옵션에서는 다음의 더 일반적인 형식을 사용할 수 있지만, 2019년 이후 출시된 리전에서는 이 형식을 사용할 수 없습니다.

```
bucket-name.s3.amazonaws.com
```

다음 값을 사용하여 버킷의 이름을 지정하지 마세요.

- Amazon S3 경로 스타일: `s3.amazonaws.com/bucket-name`
- Amazon S3 CNAME(있는 경우)

Important

버킷을 CloudFront와 함께 사용하려면 DNS 명명 요구 사항을 준수하는 이름을 사용해야 합니다. 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 버킷 규제 및 제한 단원을 참조하십시오.

단일 오리진 서버에 대해 여러 RTMP 배포 생성

Note

Adobe는 2020년 12월 31일까지 Flash 지원을 종료하기로 결정했습니다. 결과적으로 Amazon CloudFront는 더 이상 Adobe Flash Media Server를 지원하지 않으며 2020년 12월 31일까지 RTMP(Real-Time Messaging Protocol)를 더 이상 배포하지 않을 것입니다. 자세한 내용은 Amazon CloudFront 토론 포럼의 발표 전문을 읽어 보십시오.

일반적으로 Amazon S3 버킷당 하나의 RTMP 배포를 생성하지만, 동일한 버킷에 대해 여러 RTMP 배포를 생성하도록 선택할 수도 있습니다. 예를 들어, Amazon S3 버킷 하나에 대해 두 개의 배포가 있는 경우, 각 배포를 사용하여 단일 미디어 파일을 참조할 수 있습니다. 이 경우, 오리진 서버에 `media.flv`라는 미디어 파일이 있다고 한다면, CloudFront에서는 개별 `media.flv` 객체를 참조하는 것처럼 각 배포를 처리합니다. 즉, 하나의 `media.flv`를 둘 중 한 배포를 통해 액세스할 수 있으며, 또 다른 `media.flv`를 나머지 배포를 통해 액세스할 수 있습니다.

Crossdomain.xml을 사용한 액세스 제한

Note

Adobe는 2020년 12월 31일까지 Flash 지원을 종료하기로 결정했습니다. 결과적으로 Amazon CloudFront는 더 이상 Adobe Flash Media Server를 지원하지 않으며 2020년 12월 31일까지 RTMP(Real-Time Messaging Protocol)를 더 이상 배포하지 않을 것입니다. 자세한 내용은 Amazon CloudFront 토론 포럼의 발표 전문을 읽어 보십시오.

Adobe Flash Media Server `crossdomain.xml` 파일은 특정 도메인의 미디어 파일을 액세스할 수 있는 도메인을 지정합니다. CloudFront에서는 전체 도메인에서 RTMP 배포의 미디어 파일에 액세스할 수 있도록 기본 파일을 제공하며, 이 동작은 변경할 수 없습니다. 보다 제한적인 `crossdomain.xml` 파일이 Amazon S3 버킷에 포함되어 있는 경우, CloudFront에서는 이를 무시합니다.

RTMP 배포 관련 오류 코드

Note

Adobe는 2020년 12월 31일까지 Flash 지원을 종료하기로 결정했습니다. 결과적으로 Amazon CloudFront는 더 이상 Adobe Flash Media Server를 지원하지 않으며 2020년 12월 31일까지 RTMP(Real-Time Messaging Protocol)를 더 이상 배포하지 않을 것입니다. 자세한 내용은 Amazon CloudFront 토론 포럼의 발표 전문을 읽어 보십시오.

다음 표에는 CloudFront에서 미디어 플레이어에 전송할 수 있는 오류 코드가 나와 있습니다. 오류는 `Event.info.application.message` 또는 `Event.info.description`으로 반환되는 문자열의 일부입니다.

오류	설명
`DistributionNotFound`	배포를 찾을 수 없습니다.
`DistributionTypeMismatch`	이 배포는 RTMP 배포가 아닙니다.
`InvalidInstance`	인스턴스가 잘못되었습니다.

오류	설명
`InvalidURI`	URI가 잘못되었습니다.

RTMP 배포 문제 해결

> **Note**
>
> Adobe는 2020년 12월 31일까지 Flash 지원을 종료하기로 결정했습니다. 결과적으로 Amazon CloudFront는 더 이상 Adobe Flash Media Server를 지원하지 않으며 2020년 12월 31일까지 RTMP(Real-Time Messaging Protocol)를 더 이상 배포하지 않을 것입니다. 자세한 내용은 Amazon CloudFront 토론 포럼의 발표 전문을 읽어 보십시오.

미디어 파일 재생에 문제가 발생한 경우 다음 항목을 확인하세요.

확인할 항목	설명
미디어 플레이어 파일 및 미디어 파일용 개별 배포	미디어 플레이어는 정규 HTTP 배포(예: 도메인 이름 d111111abcdef8.cloudfront.net)로 제공되어야 하며, 미디어 파일은 RTMP 배포(예: 도메인 이름 s5c39gqb8ow64r.cloudfront.net)로 제공되어야 합니다. 둘 모두에 동일한 배포를 사용하지 않아야 합니다.
파일 경로의 `/cfx/st`	파일의 경로에 `/cfx/st`가 포함되어 있는지 확인합니다. Amazon S3 버킷의 객체에 대한 경로에는 `/cfx/st`가 포함될 필요가 없습니다. 자세한 내용은 미디어 플레이어 구성 (p. 277) 단원을 참조하십시오.
파일 경로의 파일 이름	일부 미디어 플레이어에서는 파일 이름 확장명(예: `mp4:`)을 파일 경로의 파일 이름 앞에 포함하도록 요구합니다. 또한 일부 미디어 플레이어에서는 파일 이름 확장명(예: `.mp4`)을 파일 경로에서 제외하도록 요구합니다. 자세한 내용은 MPEG 파일 (p. 278) 단원을 참조하십시오. **Note** Amazon S3 버킷에 있는 미디어 파일의 이름에는 항상 해당 파일 이름 확장명을 포함해야 합니다.
방화벽의 포트 1935	Adobe Flash Media Server에서는 RTMP에 포트 1935를 사용합니다. 방화벽에서 이 포트가 열려 있어야 합니다. 그렇지 않은 경우, 일반적으로 반환되는 메시지는 "비디오를 재생할 수 없음"입니다. 또한 RTMPT로 전환하고 포트 80을 사용하여 HTTP를 통해 터널링할 수도 있습니다.
Adobe Flash Player 메시징	기본적으로 Adobe Flash Player에서는 재생을 시도 중인 비디오 파일이 없는 경우 메시지를 표시하지 않습니다. 그 대신 파일이 나타날 때까지 기다립니다. 최종 사용자의 사용 환경을 개선하기 위해 이 동작을 변경하려 할 수 있습니다. 비디오가 없는 경우 기다리는 대신 플레이어에서 메시지를 보내게 하려면 `play("vid",0,-1)` 대신 `play("vid")`을 사용합니다.

Lambda@Edge를 사용하여 엣지에서 콘텐츠 사용자 지정

Lambda@Edge는 CloudFront를 통해 전달되는 콘텐츠를 사용자 지정하는 함수를 실행할 수 있게 해주는 AWS Lambda가 확장된 컴퓨팅 서비스입니다. 서버를 프로비저닝하거나 관리하지 않고 한 리전 US-East-1(버지니아 북부)에서 Node.js 또는 Python 함수를 작성한 후 최종 사용자에게 가까운 전 세계 AWS 위치에서 해당 함수를 실행할 수 있습니다. Lambda@Edge는 하루 몇 번의 요청에서 초당 수천 개의 요청으로 자동 확장됩니다. 오리진 서버가 아니라 최종 사용자에게 가까운 AWS 위치에서 요청을 처리하므로 지연 시간이 크게 단축되고 사용자 경험이 상당히 개선됩니다.

CloudFront 배포를 Lambda@Edge 함수와 연결하면 CloudFront가 CloudFront 엣지 로케이션에서 요청 및 응답을 가로챕니다. 다음과 같은 CloudFront 이벤트가 발생할 때 Lambda 함수를 실행할 수 있습니다.

- CloudFront가 최종 사용자로부터 요청을 수신할 때(최종 사용자 요청)
- CloudFront가 오리진에 요청을 전달하기 전(오리진 요청)
- CloudFront가 오리진으로부터 응답을 수신할 때(오리진 응답)
- CloudFront가 최종 사용자에게 응답을 반환하기 전(최종 사용자 응답)

Lambda@Edge 처리는 여러 용도로 사용됩니다. 예:

- Lambda 함수는 사용자가 A/B 테스트를 위해 사이트의 다양한 버전을 볼 수 있도록 쿠키를 검사하고 URL을 다시 작성합니다.
- CloudFront는 `User-Agent` 헤더를 확인하여 사용 중인 디바이스를 기반으로 디바이스에 대한 정보를 포함하여 최종 사용자에게 다양한 객체를 반환할 수 있습니다. 예를 들어, CloudFront는 디바이스의 화면 크기에 따라 다른 이미지를 반환할 수 있습니다. 마찬가지로 함수는 `Referer` 헤더 값을 고려하여 CloudFront가 사용 가능한 최저 해상도의 이미지를 봇에 반환하게 할 수 있습니다.
- 아니면 다른 기준으로 쿠키를 검사할 수 있습니다. 예를 들어, 의류를 판매하는 소매 웹 사이트에서 사용자가 재킷에 대해 선택한 색상을 표시하기 위해 쿠키를 사용하는 경우 Lambda 함수는 CloudFront가 선택된 색상으로 재킷 이미지를 반환하도록 요청을 변경할 수 있습니다.
- Lambda 함수는 CloudFront 최종 사용자 요청 또는 오리진 요청 이벤트가 발생할 때 HTTP 응답을 생성할 수 있습니다.
- 함수는 헤더 또는 권한 부여 토큰을 검사하고, CloudFront가 오리진으로 요청을 전달하기 전에 헤더를 삽입하여 콘텐츠에 대한 액세스 권한을 제어할 수 있습니다.
- 또한 Lambda 함수는 외부 리소스에 대한 네트워크 호출을 생성하여 사용자 자격 증명을 확인하거나 추가 콘텐츠를 가져와 응답을 사용자 지정할 수 있습니다.

샘플 코드 및 추가 예제는 Lambda@Edge 함수 예제 (p. 330) 단원을 참조하십시오.

주제

- Lambda@Edge 함수 생성 및 사용 시작하기 (p. 282)
- Lambda@Edge에 대한 IAM 권한 및 역할 설정 (p. 292)
- Lambda@Edge 함수 작성 및 생성 (p. 298)
- Lambda@Edge 함수에 대한 트리거 추가 (p. 302)
- Lambda@Edge 함수 테스트 및 디버깅 (p. 306)

- Lambda 함수용 CloudWatch 지표 및 CloudWatch Logs (p. 314)
- Lambda@Edge 함수 및 복제본 삭제 (p. 315)
- Lambda@Edge 이벤트 구조 (p. 315)
- 요청 및 응답으로 작업 (p. 326)
- Lambda@Edge 함수 예제 (p. 330)
- Lambda 함수에 대한 요구 사항 및 제한 사항 (p. 357)

Lambda@Edge 함수 생성 및 사용 시작하기

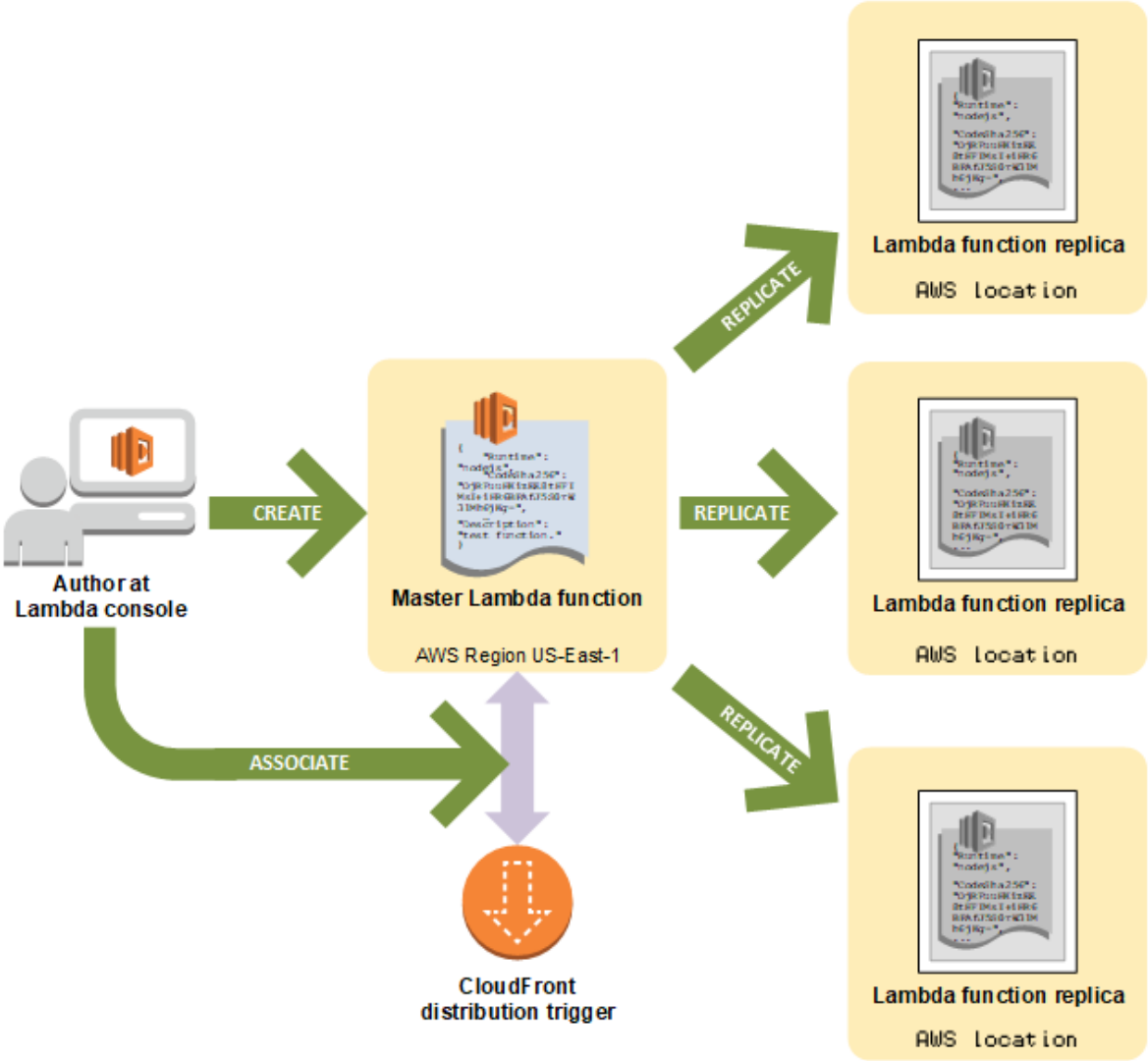
Lambda@Edge 함수로 여러 가지 유용한 작업을 할 수 있지만, 시작 과정이 약간 복잡하게 느껴질 수 있습니다. 이 단원에서는 Lambda@Edge가 CloudFront와 작동하는 방법을 개괄하고 간단한 예제 단계를 통해 자습서를 안내합니다.

> Tip
>
> Lambda@Edge의 작동 방식을 숙지하고 Lambda@Edge 함수를 작성한 뒤, Lambda@Edge를 사용하여 자체 사용자 지정 솔루션을 만드는 방법을 자세히 알아봅니다. 함수 생성 및 업데이트, 이벤트 구조 및 CloudFront 트리거 추가에 대해 자세히 알아봅니다. 또한 Lambda@Edge 함수 예제 (p. 330)에서 다양한 아이디어와 코드 샘플도 다룹니다.

다음은 CloudFront에서 Lambda 함수를 생성 및 사용하는 방법에 대한 개요입니다.

1. AWS Lambda 콘솔을 사용하여 미국 동부(버지니아 북부) 지역에서 Lambda 함수를 생성합니다. (아니면 예를 들어 AWS SDK 중 하나를 사용하여 프로그래밍 방식으로 함수를 만들 수도 있습니다.)
2. 번호가 지정된 함수 버전을 저장하고 게시합니다.

   함수를 변경하려면 미국 동부(버지니아 북부) 지역에서 $LATEST 버전의 함수를 편집해야 합니다. 그런 다음 번호가 지정된 새 버전을 게시하고 CloudFront와 연동되도록 설정합니다.
3. 함수가 적용되는 CloudFront 배포 및 캐시 동작을 선택합니다. 그런 다음 함수를 실행시키는 하나 이상의 CloudFront 이벤트(트리거)를 지정합니다. 예를 들어, CloudFront가 최종 사용자의 요청을 받으면 함수가 실행되도록 하는 트리거를 생성할 수 있습니다.
4. 트리거를 생성하면 Lambda가 그 함수를 전 세계의 AWS 리전에 복제합니다.



주제

- 자습서: 간단한 Lambda@Edge 함수 생성 (p. 283)

자습서: 간단한 Lambda@Edge 함수 생성

이 자습서에서는 CloudFront에서 실행되는 샘플 Node.js 함수를 작성하고 추가하는 과정을 통해 Lambda@Edge 시작 방법을 보여줍니다. 여기서 살펴볼 예제에서는 응답에 HTTP 보안 헤더를 추가하여 웹 사이트의 보안 및 개인 정보 보호를 강화합니다. 이 연습에는 웹 사이트가 필요하지 않습니다. CloudFront가 파일을 검색할 때 보안 헤더를 응답에 추가하면 됩니다.

이 예제에서는 Lambda@Edge 함수를 만들고 구성하는 단계를 설명합니다. 자체 Lambda@Edge 솔루션을 만드는 경우 비슷한 단계를 따르고 동일한 옵션 중에서 선택합니다.

주제

- 1단계: AWS 계정에 가입 (p. 284)
- 2단계: CloudFront 배포 만들기 (p. 284)
- 3단계: 함수 생성 (p. 284)

- 4단계: 함수를 실행할 CloudFront 트리거 추가 (p. 288)
- 5단계: 함수 실행 확인 (p. 290)
- 6단계: 문제 해결 (p. 291)
- 7단계: 예제 리소스 정리 (p. 291)
- 추가 정보 리소스 (p. 292)

1단계: AWS 계정에 가입

아직 Amazon Web Services에 가입하지 않았다면 https://aws.amazon.com/에서 가입하십시오. Sign Up Now(지금 가입)를 선택하고 필요한 정보를 입력합니다.

2단계: CloudFront 배포 만들기

예제 Lambda@Edge 함수를 만들려면 먼저 콘텐츠를 제공할 오리진이 있는 CloudFront 환경부터 갖춰야 합니다.

CloudFront를 처음 사용하십니까? CloudFront는 엣지 로케이션의 전 세계 네트워크를 사용해 콘텐츠를 전송합니다. Lambda 함수와 함께 CloudFront를 설정하면 이 함수로 보다 관련성 있는 콘텐츠가 표시되게 사용자 지정하여 성능을 높일 수 있습니다. CloudFront에 익숙하지 않은 경우, 자습서를 진행하기 전에 잠깐 시간을 들여 간단한 개요를 읽고 CloudFront가 콘텐츠를 캐시 및 제공하는 방법을 알아보십시오.

이 예제에서는 Amazon S3 버킷을 배포의 오리진으로 사용하는 CloudFront 배포를 만듭니다. 사용할 환경이 이미 있는 경우 이 단계를 건너뛸 수 있습니다.

Amazon S3 오리진으로 CloudFront 배포를 생성하려면

1. 이미지 파일 등 샘플 콘텐츠가 될 파일 한두 개로 Amazon S3 버킷을 만듭니다. Amazon S3에 콘텐츠 업로드의 단계를 따르면 됩니다. 버킷의 객체에 대한 퍼블릭 읽기 액세스를 허용하는 권한을 설정해야 합니다.
2. CloudFront 웹 배포 만들기의 단계에 따라 CloudFront 배포를 만들고 오리진으로 S3 버킷을 추가합니다. 이미 배포가 있다면 버킷을 그 배포의 오리진으로 추가할 수 있습니다.

   Tip

   배포 ID를 기록해 둡니다. 이 자습서 후반부에서 함수의 CloudFront 트리거를 추가할 때 드롭다운 목록에서 배포 ID를 선택해야 합니다(예: E653W22221KDDL).

3단계: 함수 생성

이 단계에서는 Lambda 콘솔에 있는 블루프린트 템플릿으로 시작하여 Lambda 함수를 생성합니다. 이 함수는 CloudFront 배포의 보안 헤더를 업데이트하는 코드를 추가합니다.

Lambda 또는 Lambda@Edge를 처음 사용하십니까? Lambda@Edge에서 CloudFront 트리거를 사용하여 Lambda 함수를 호출할 수 있습니다. CloudFront 배포를 Lambda 함수와 연결하면 CloudFront가 CloudFront 엣지 로케이션에서 요청 및 응답을 가로채고 함수를 실행합니다. Lambda 함수로 보안을 강화하거나 최종 사용자에게 더 관련성 있는 정보가 표시되도록 사용자 지정하여 성능을 높일 수 있습니다. 이 자습서에서 만드는 함수는 CloudFront 응답의 보안 헤더를 업데이트합니다.

Lambda 함수를 만들려면 몇 가지 단계를 거쳐야 합니다. 이 자습서에서는 블루프린트 템플릿을 함수의 토대로 사용한 다음, 보안 헤더를 설정하는 코드로 함수를 업데이트합니다. 마지막으로 함수를 실행할 CloudFront 트리거를 추가하고 배포합니다.

Lambda 함수를 만들려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/lambda/에서 AWS Lambda 콘솔을 엽니다.

Important

현재 US-East-1(버지니아 북부) 리전(us-east-1)에 있는지 확인합니다. 이 리전에 있어야 Lambda@Edge 함수를 만들 수 있습니다.

2. 함수 생성을 선택합니다.
3. 함수 생성 페이지에서 블루프린트 사용을 선택한 다음 검색 필드에 **`cloudfront`**를 입력하여 CloudFront 블루프린트를 필터링합니다. Keyword : cloudfront가 표시되고, CloudFront 태그가 지정된 블루프린트가 모두 나열됩니다.

   Note

   CloudFront 블루프린트는 US-East-1(버지니아 북부) 리전(us-east-1)에서만 사용 가능합니다.

4. cloudfront-modify-response-header 블루프린트를 함수의 템플릿으로 선택합니다.
5. 다음과 같이 함수에 대한 정보를 입력합니다.

   이름

   함수 이름을 입력합니다.

   실행 역할

   함수에 대한 권한 설정 방법을 선택합니다. 권장되는 기본 Lambda@Edge 권한 정책 템플릿을 사용하려면 AWS 정책 템플릿에서 새 역할 생성을 선택합니다.

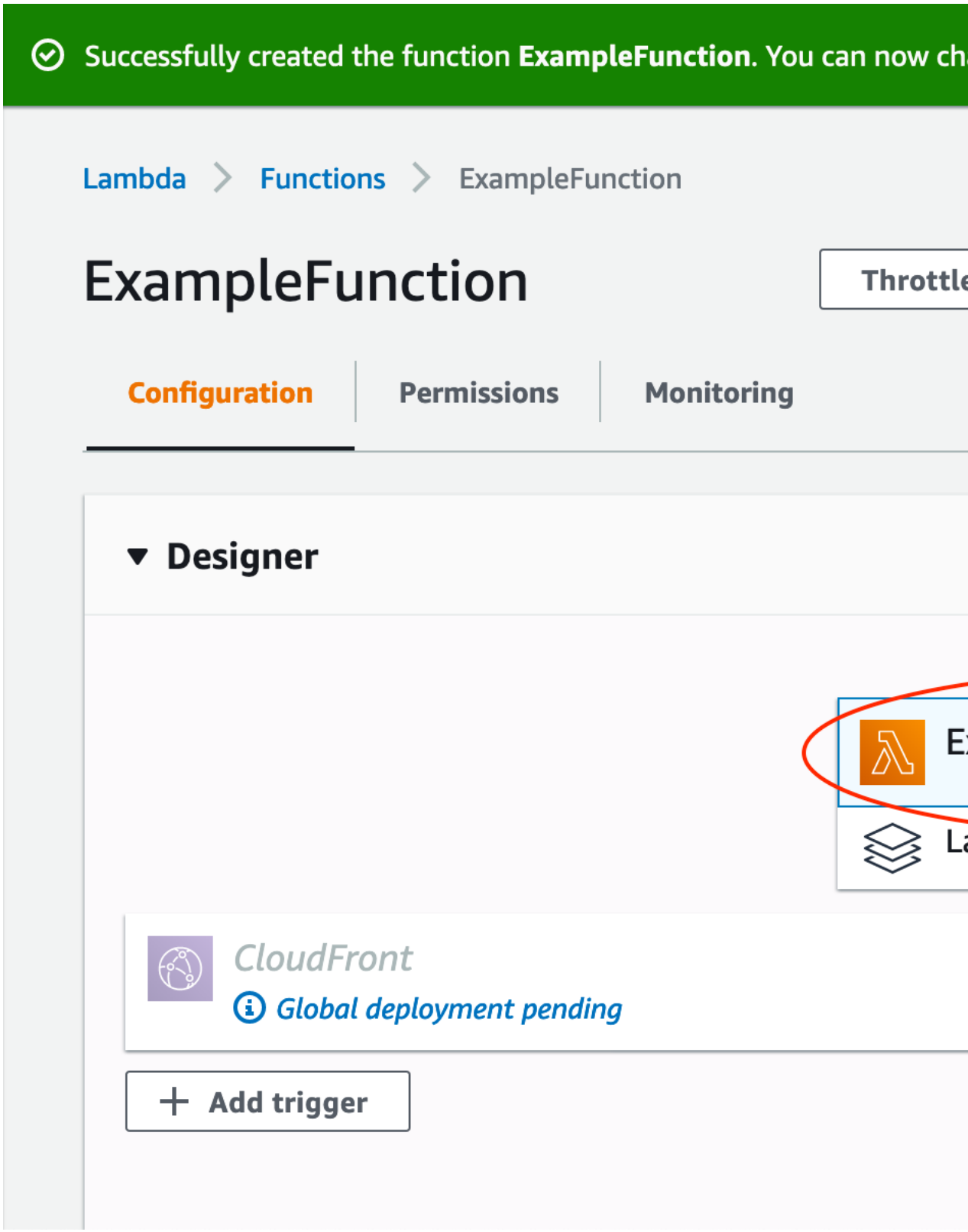
   역할 이름

   정책 템플릿이 생성하는 역할 이름을 입력합니다.

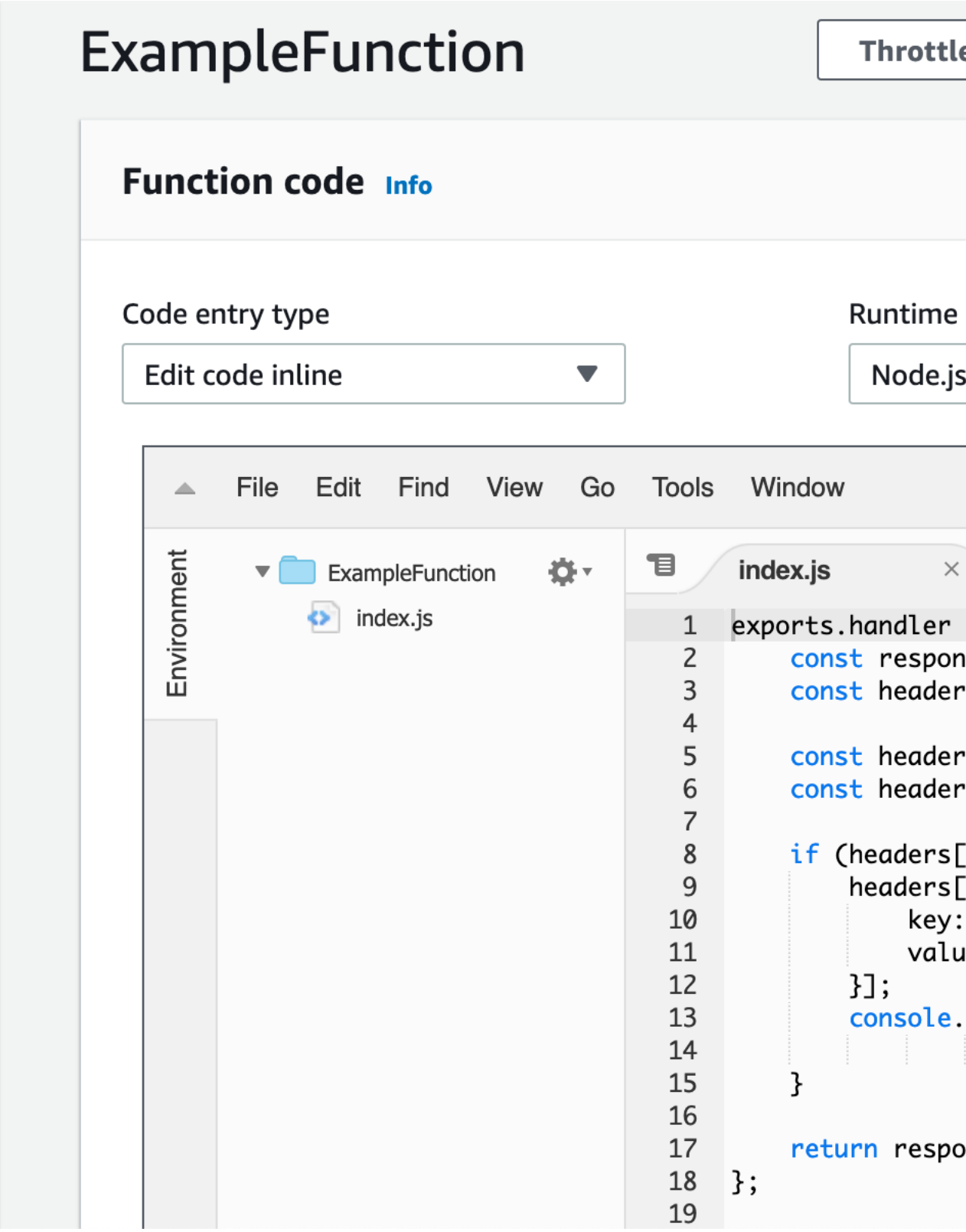
   정책 템플릿

   CloudFront 블루프린트를 함수의 기본으로 선택했으므로 Lambda는 정책 템플릿 기본 Edge Lambda 권한을 자동으로 추가합니다. 이 정책 템플릿은 CloudFront이(가) 전 세계 CloudFront 로케이션에서 이 Lambda 함수를 실행하도록 하는 실행 역할 권한을 추가합니다. 자세한 내용은 Lambda@Edge에 대한 IAM 권한 및 역할 설정 (p. 292) 단원을 참조하십시오.

6. 함수 생성을 선택합니다. Lambda가 함수를 만들면 다음 페이지에서 함수 구성을 확인할 수 있습니다.
7. 해당 페이지의 Designer(디자이너) 섹션에서 다음 그림과 같이 함수 이름을 선택합니다. 이 예제에서 함수 이름은 ExampleFunction입니다.



8. 다음 그림과 같이 페이지를 아래로 스크롤하여 Function code(함수 코드) 섹션을 찾습니다.


ExampleFunction
Throttle
Function code Info
Code entry type
Edit code inline
Runtime
Node.js
File Edit Find View Go Tools Window
Environment
ExampleFunction
index.js
index.js
exports.handler
const respon
const header
const header
const header
if (headers[
headers[
key:
valu
console.
}
return respo
};

템플릿 코드를 오리진에서 반환하는 보안 헤더를 수정하는 함수로 바꿉니다. 예를 들면 다음과 비슷한 코드를 사용할 수 있습니다.

```
'use strict';
exports.handler = (event, context, callback) => {

    //Get contents of response
    const response = event.Records[0].cf.response;
    const headers = response.headers;

    //Set new headers
    headers['strict-transport-security'] = [{key: 'Strict-Transport-Security', value:
 'max-age= 63072000; includeSubdomains; preload'}];
    headers['content-security-policy'] = [{key: 'Content-Security-Policy', value:
 "default-src 'none'; img-src 'self'; script-src 'self'; style-src 'self'; object-src
 'none'"}];
    headers['x-content-type-options'] = [{key: 'X-Content-Type-Options', value:
 'nosniff'}];
    headers['x-frame-options'] = [{key: 'X-Frame-Options', value: 'DENY'}];
    headers['x-xss-protection'] = [{key: 'X-XSS-Protection', value: '1; mode=block'}];
    headers['referrer-policy'] = [{key: 'Referrer-Policy', value: 'same-origin'}];

    //Return modified response
    callback(null, response);
};
```

9. 저장을 선택하여 업데이트된 코드를 저장합니다.

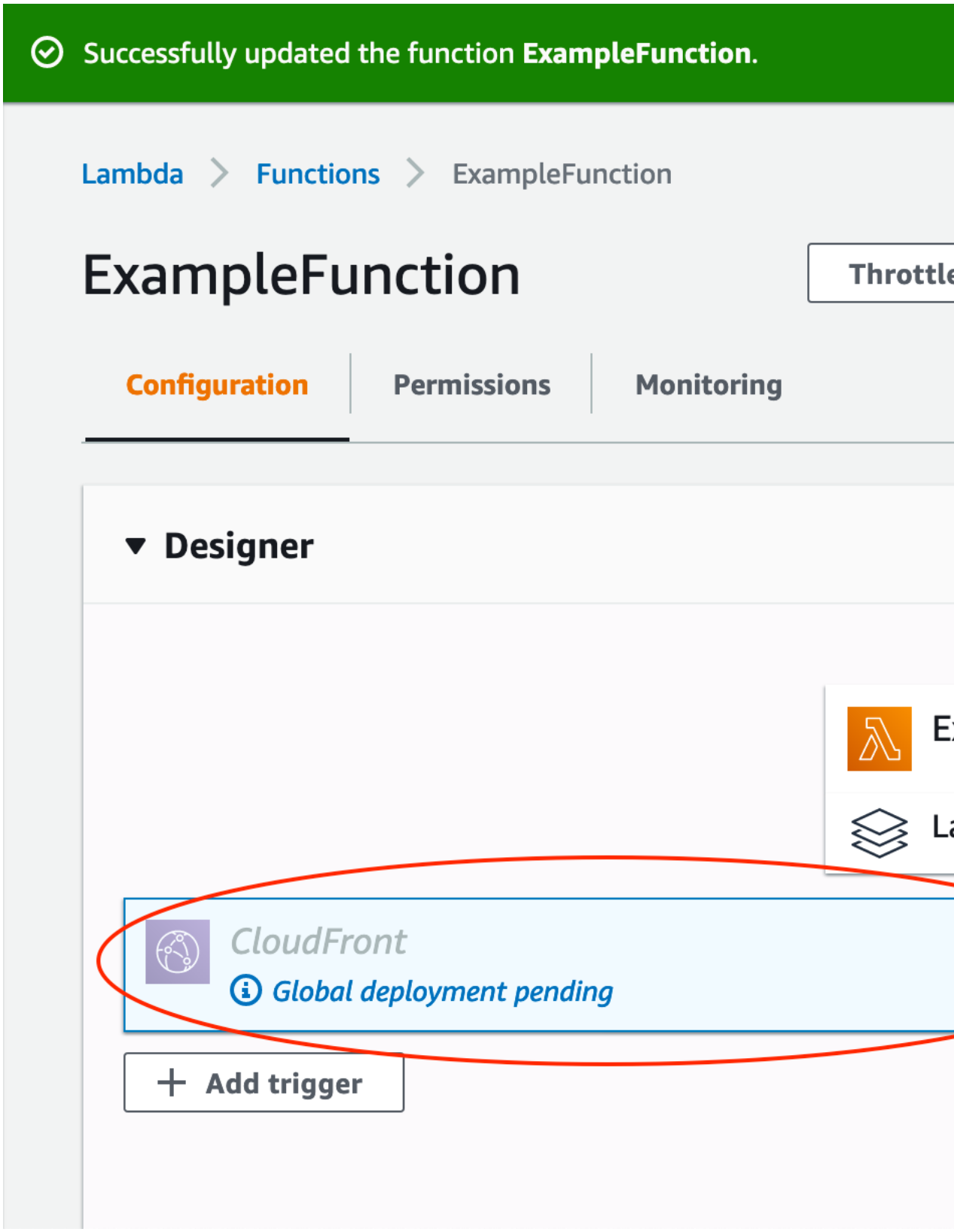
다음 섹션으로 이동하여 함수를 실행할 CloudFront 트리거를 추가합니다.

4단계: 함수를 실행할 CloudFront 트리거 추가

이제 보안 헤더를 업데이트할 Lambda 함수가 생성되었으므로, 그 함수를 실행하여 오리진에서 해당 배포에 대해 CloudFront(으)로 보내는 모든 응답에 헤더를 추가할 CloudFront 트리거를 구성해야 합니다.

함수의 CloudFront 트리거를 구성하려면

1. 해당 페이지의 Designer(디자이너) 섹션에서 다음 그림과 같이 CloudFront를 선택합니다.


Successfully updated the function ExampleFunction.
Lambda
Functions
ExampleFunction
ExampleFunction
Throttle
Configuration
Permissions
Monitoring
Designer
CloudFront
Global deployment pending
Add trigger

2. 페이지를 아래로 스크롤하여 Configure triggers(트리거 구성) 섹션을 찾은 다음 Deploy to Lambda@Edge(Lambda@Edge로 배포)를 선택합니다.
3. Deploy to Lambda@Edge(Lambda@Edge로 배포) 페이지에서 CloudFront 트리거 구성 아래에 다음 정보를 입력합니다.

   Distribution

   함수와 연결할 CloudFront 배포 ID입니다. 드롭다운 목록에서 배포 ID를 선택합니다.

   캐시 동작

   트리거에 사용할 캐시 동작입니다. 이 예제에서는 *로 설정된 값을 그대로 둡니다. 이 값은 해당 배포의 기본 캐시 동작을 의미합니다. 자세한 내용은 캐시 동작 설정 (p. 43) 주제에서 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.

   CloudFront 이벤트

   언제 함수를 실행할지 지정하는 트리거입니다. 여기서는 CloudFront가 오리진에서 응답을 반환할 때마다 보안 헤더 함수가 실행되도록 하겠습니다. 드롭다운 목록에서 오리진 응답을 선택합니다. 자세한 내용은 Lambda@Edge 함수에 대한 트리거 추가 (p. 302) 단원을 참조하십시오.
4. Confirm deploy to Lambda@Edge(Lambda@Edge로 배포 확인)에서 확인란을 선택하여 트리거가 배포되어 모든 AWS 위치에서 함수를 실행함을 확인합니다.
5. 배포를 선택하여 트리거를 추가하고 함수를 전 세계 AWS 위치에 복제합니다. 그러면 Deploy to Lambda@Edge(Lambda@Edge로 배포) 페이지를 닫아도 됩니다.
6. 함수가 복제될 때까지 기다립니다. 일반적으로 몇 분 정도 걸립니다.

   CloudFront 콘솔로 이동하여 해당 배포를 보고 복제가 완료되었는지 확인할 수 있습니다. 배포 상태가 진행 중에서 다시 배포 완료로 바뀌었는지 확인합니다. 이것은 함수가 복제되었다는 의미입니다. 함수가 작동하는지 확인하려면 다음 단원의 단계를 따릅니다.

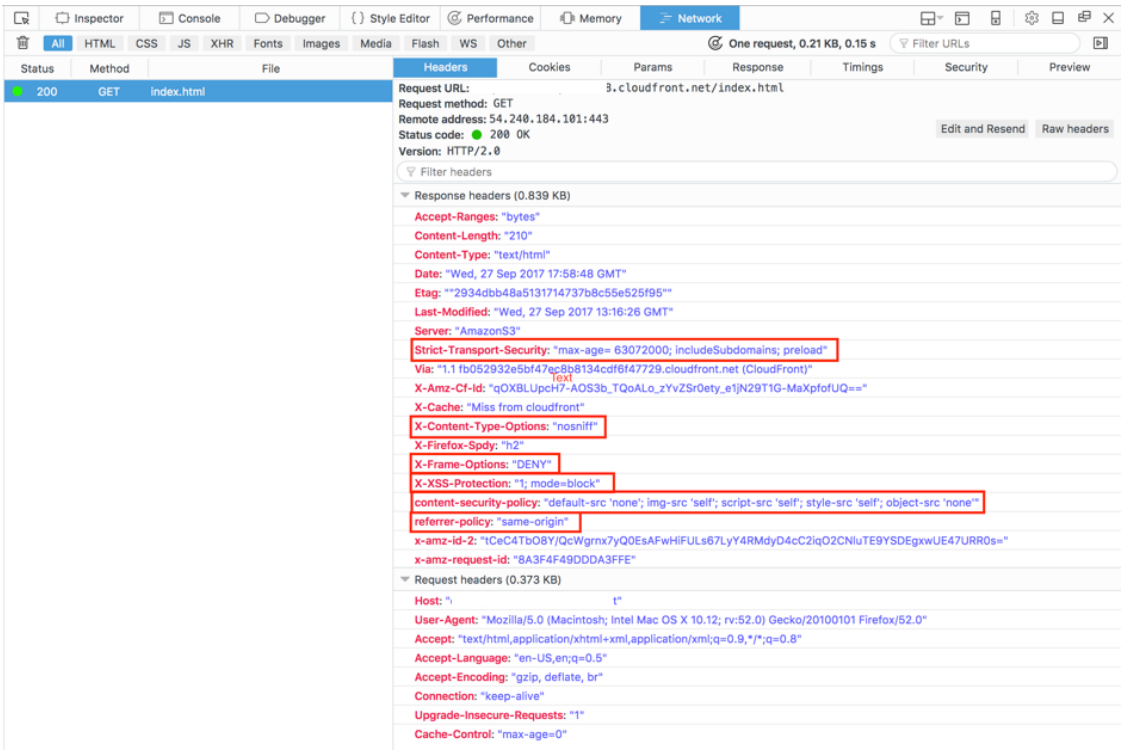
5단계: 함수 실행 확인

Lambda 함수를 만들고 CloudFront 배포에 대해 이를 실행하는 트리거를 구성했으니 이제 이 함수가 원하는 결과를 달성하는지 확인해야 합니다. 이 예제에서는 CloudFront가 반환하는 HTTP 헤더를 확인하여 보안 헤더가 추가되었는지 알아봅니다.

Lambda@Edge 함수가 보안 헤더를 추가하는지 확인하려면

1. 브라우저에서 S3 버킷에 있는 파일의 URL을 입력합니다. 예를 들면 `http://d111111abcdef8.cloudfront.net/image.jpg`와 비슷한 URL을 사용할 수 있습니다.

   파일 URL에 사용할 CloudFront 도메인 이름에 대한 자세한 내용은 CloudFront에서 파일에 대한 URL 형식 사용자 지정 (p. 79) 단원을 참조하십시오.
2. 브라우저에서 웹 개발자 도구 모음을 엽니다. 예를 들어 Chrome 브라우저 창에서는 컨텍스트(마우스 오른쪽 버튼 클릭) 메뉴를 연 다음 Inspect(검사)를 선택합니다.
3. 네트워크 탭을 선택합니다.
4. 페이지를 새로 고쳐 이미지를 확인한 다음 왼쪽 창에서 HTTP 요청을 선택합니다. HTTP 헤더가 별도의 창에 표시됩니다.
5. HTTP 헤더 목록을 살펴보면서 원하는 보안 헤더가 목록에 포함되어 있는지 확인합니다. 예를 들어 다음 스크린샷에 표시된 것과 비슷한 헤더가 보일 것입니다.



추가한 보안 헤더가 헤더 목록에 있으면 완료된 것입니다. 첫 번째 Lambda@Edge 함수가 생성되었습니다. CloudFront에서 오류를 반환하거나 다른 문제가 있는 경우에는 계속해서 다음 단계로 넘어가 문제를 해결하십시오.

6단계: 문제 해결

CloudFront가 오류를 반환하거나 원하는 보안 헤더를 추가하지 않는 경우, CloudWatch 로그를 조사하여 함수의 실행 상태를 확인할 수 있습니다. 함수를 실행한 위치와 가장 가까운 AWS 위치에 저장된 로그를 사용해야 합니다.

예를 들어 런던에서 파일을 볼 때는 CloudWatch 콘솔에서 리전을 유럽(런던)로 변경해야 합니다.

Lambda@Edge 함수에 대한 CloudWatch 로그를 검사하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudwatch/에서 CloudWatch 콘솔을 엽니다.
2. 리전을 브라우저에서 파일을 볼 때 표시되는 위치로 변경합니다. 이는 함수가 실행되고 있는 리전입니다.
3. 왼쪽 창에서 로그를 선택하여 해당 배포의 로그를 봅니다.

자세한 정보는 Amazon CloudWatch를 사용하여 CloudFront 모니터링 (p. 416) 단원을 참조하십시오.

7단계: 예제 리소스 정리

이 자습서에 사용할 용도로만 Amazon S3 버킷과 CloudFront 배포를 만들었다면 할당한 AWS 리소스를 삭제하여 더 이상 요금이 발생하지 않도록 하십시오. AWS 리소스를 삭제한 뒤에는 추가한 콘텐츠를 이용할 수 없게 됩니다.

작업

- S3 버킷 삭제 (p. 292)
- CloudFront 배포 삭제 (p. 292)

S3 버킷 삭제

Amazon S3 버킷을 삭제하기 전에 버킷에 대한 로깅이 비활성화되어 있는지 확인하십시오. 그렇지 않으면 삭제해도 AWS가 해당 버킷에 계속 로그를 기록합니다.

버킷에 대한 로깅을 비활성화하려면

1. https://console.aws.amazon.com/s3/에서 Amazon S3 콘솔을 엽니다.
2. 버킷을 선택한 다음, 속성를 선택합니다.
3. 속성에서 로깅을 선택합니다.
4. 활성화 확인란의 선택을 취소합니다.
5. Save를 선택합니다.

이제 버킷을 삭제할 수 있습니다. 자세한 내용은 Amazon Simple Storage Service Console 사용 설명서의 S3 버킷을 삭제하려면 어떻게 해야 합니까? 단원을 참조하십시오.

CloudFront 배포 삭제

CloudFront 배포를 삭제하기 전에 반드시 배포를 비활성화해야 합니다. 비활성화된 배포가 작동하지 않아 요금이 발생하지 않습니다. 언제든지 비활성화된 배포를 활성화할 수 있습니다. 비활성화된 배포를 삭제한 뒤에는 사용할 수 없습니다.

CloudFront 배포를 비활성화 및 삭제하는 방법

1. https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 비활성화하려는 배포를 선택한 후 비활성화를 선택합니다.
3. 확인 메시지가 표시되면 예, 비활성화를 선택합니다.
4. 비활성화된 배포를 선택한 후 삭제를 선택합니다.
5. 확인 메시지가 나타나면 예, 삭제합니다를 선택합니다.

추가 정보 리소스

Lambda@Edge 함수의 작동 방식에 대한 기본적인 내용을 배웠습니다. 이제 다음 자료를 읽고 자세히 알아보십시오.

- Lambda@Edge 함수 예제 (p. 330)
- Lambda@Edge 설계 모범 사례
- Lambda@Edge로 지연 시간 감소 및 컴퓨팅 작업을 Edge로 이전

Lambda@Edge에 대한 IAM 권한 및 역할 설정

Lambda@Edge를 구성하려면 특정 IAM 권한과 IAM 실행 역할을 설정해야 합니다. Lambda@Edge는 Lambda 함수를 CloudFront 리전에 복제하고 CloudWatch에서 CloudFront 로그 파일을 사용할 수 있도록 하기 위해 서비스 연결 역할도 생성합니다.

주제

- Lambda 함수를 CloudFront 배포와 연결하는 데 필요한 IAM 권한 (p. 293)

- 서비스 보안 주체에 대한 함수 실행 역할 (p. 293)
- Lambda@Edge의 서비스 연결 역할 (p. 294)

Lambda 함수를 CloudFront 배포와 연결하는 데 필요한 IAM 권한

IAM 사용자가 Lambda 함수를 CloudFront 배포와 연결하려면 AWS Lambda,를 사용할 때 필요한 IAM 권한 외에 다음과 같은 IAM 권한이 필요합니다.

- `lambda:GetFunction`

  사용자는 Lambda 함수가 포함된 .zip 파일을 다운로드하기 위한 미리 서명된 URL과 Lambda 함수에 대한 구성 정보를 가져올 수 있습니다.

  리소스의 경우, 다음 예제와 같이 CloudFront 이벤트가 발생할 때 실행할 함수 버전의 ARN을 지정합니다.

  ```
  arn:aws:lambda:us-east-1:123456789012:function:TestFunction:2
  ```
- `lambda:EnableReplication*`

  함수 코드 및 구성을 가져올 수 있는 Lambda 복제 서비스 권한을 부여하는 리소스 정책에 권한을 추가합니다.

  리소스의 경우, 다음 예제와 같이 CloudFront 이벤트가 발생할 때 실행할 함수 버전의 ARN을 지정합니다.

  ```
  arn:aws:lambda:us-east-1:123456789012:function:TestFunction:2
  ```
- `iam:CreateServiceLinkedRole`

  사용자는 Lambda@Edge에서 사용되는 서비스 연결 역할을 생성하여 CloudFront에서 Lambda 함수를 복제할 수 있습니다. Lambda@Edge에서 사용하는 첫 번째 배포에서 이 역할이 생성되고 나면 Lambda@Edge에서 사용하는 다른 배포에 대한 권한을 추가할 필요가 없습니다.
- `cloudfront:UpdateDistribution` 또는 `cloudfront:CreateDistribution`

  `cloudfront:UpdateDistribution`을 사용하여 배포를 업데이트하거나 `cloudfront:CreateDistribution`을 사용하여 배포를 생성합니다.

자세한 내용은 다음 설명서를 참조하십시오.

- CloudFront의 Identity and Access Management (p. 428) 이 가이드의 내용.
- AWS Lambda Developer Guide의 AWS Lambda 인증 및 액세스 제어

서비스 보안 주체에 대한 함수 실행 역할

서비스 보안 주체 `lambda.amazonaws.com` 및 `edgelambda.amazonaws.com`가 수임할 수 있는 IAM 역할을 생성해야 합니다. 이 역할은 서비스 보안 주체가 사용자의 함수를 실행할 때 수임합니다. 자세한 내용은 IAM 사용 설명서의 "직무 기능에 관한 AWS 관리형 정책" 주제에 나온 역할 생성 및 정책 연결(콘솔) 단원을 참조하십시오.

IAM의 신뢰 관계 탭 아래에 이 역할을 추가합니다(권한 탭 아래에는 추가하지 마십시오).

다음은 역할 신뢰 정책의 예입니다.

```
{
   "Version": "2012-10-17",
```

```
   "Statement": [
      {
         "Effect": "Allow",
         "Principal": {
            "Service": [
               "lambda.amazonaws.com",
               "edgelambda.amazonaws.com"
            ]
         },
         "Action": "sts:AssumeRole"
      }
   ]
}
```

실행 역할에 부여해야 하는 권한에 대한 자세한 내용은 AWS Lambda Developer Guide에 있는 권한 관리: IAM 역할(실행 역할) 사용 단원을 참조하십시오. 다음을 참조하십시오.

- 기본적으로 CloudFront 이벤트가 Lambda 함수를 트리거할 때마다 데이터는 CloudWatch Logs에 기록됩니다. 이러한 로그를 사용하려면 실행 역할에 CloudWatch Logs에 데이터를 기록할 권한이 있어야 합니다. 사전 정의된 AWSLambdaBasicExecutionRole을 사용하여 실행 역할에 대한 권한을 부여할 수 있습니다.

  For more information about CloudWatch Logs, see Lambda 함수용 CloudWatch 지표 및 CloudWatch Logs (p. 314).
- Lambda 함수 코드가 S3 버킷에서 객체를 읽는 것처럼 다른 AWS 리소스에 액세스하는 경우 실행 역할에는 해당 작업을 수행할 수 있는 권한이 필요합니다.

Lambda@Edge의 서비스 연결 역할

Lambda@Edge에서는 AWS Identity and Access Management(IAM) 서비스 연결 역할을 사용합니다. 서비스 연결 역할은 서비스에 직접 연결된 고유한 유형의 IAM 역할입니다. 서비스 연결 역할은 해당 서비스에서 사전 정의하며 서비스에서 사용자를 대신하여 다른 AWS 서비스를 호출하기 위해 필요한 모든 권한을 포함합니다.

Lambda@Edge는 다음 IAM 서비스 연결 역할을 사용합니다.

- AWSServiceRoleForLambdaReplicator–Lambda@Edge에서는 이 역할을 사용해 Lambda@Edge가 AWS 리전에 함수를 복제하도록 합니다.
- AWSServiceRoleForCloudFrontLogger–CloudFront에서는 Lambda@Edge 확인 오류를 디버깅하기 위해 이 역할을 사용해 CloudWatch 계정으로 로그 파일을 푸시합니다.

CloudFront에 Lambda@Edge 트리거를 처음 추가할 때 AWSServiceRoleForLambdaReplicator라는 역할이 자동으로 생성되어 Lambda@Edge가 AWS 리전에 함수를 복제할 수 있도록 합니다. Lambda@Edge 함수를 사용하려면 이 역할이 필요합니다. AWSServiceRoleForLambdaReplicator 역할의 ARN의 모양은 다음과 같습니다.

```
arn:aws:iam::123456789012:role/aws-service-role/
replicator.lambda.amazonaws.com/AWSServiceRoleForLambdaReplicator
```

AWSServiceRoleForCloudFrontLogger라는 두 번째 역할은 CloudFront에서 Lambda@Edge 오류 로그 파일을 CloudWatch로 푸시하도록 허용하기 위해 Lambda@Edge 함수 연결을 추가한 경우 자동으로 생성됩니다. AWSServiceRoleForCloudFrontLogger 역할의 ARN의 모양은 다음과 같습니다.

```
arn:aws:iam::account_number:role/aws-service-role/
logger.cloudfront.amazonaws.com/AWSServiceRoleForCloudFrontLogger
```

서비스 연결 역할이 있으면 필요한 권한을 수동으로 추가할 필요가 없으므로 Lambda@Edge를 설정 및 사용하기가 쉽습니다. 서비스 연결 역할의 권한은 Lambda@Edge에서 정의하며, Lambda@Edge만이 그 역할을

맡을 수 있습니다. 정의된 권한에는 신뢰 정책과 권한 정책이 포함됩니다. 권한 정책은 다른 어떤 IAM 엔터티에도 연결할 수 없습니다.

서비스 연결 역할을 삭제하려면 먼저 연결된 CloudFront 또는 Lambda@Edge 리소스를 제거해야 합니다. 그러면 활성 리소스에 액세스하는 데 필요한 서비스 연결 역할을 제거할 필요가 없어 Lambda@Edge 리소스를 보호할 수 있습니다.

서비스 연결 역할을 지원하는 기타 서비스에 대한 자세한 정보는 IAM으로 작업하는 AWS 서비스를 참조하고 서비스 연결 역할 열에 예가 있는 서비스를 찾으십시오.

Lambda@Edge의 서비스 연결 역할 권한

Lambda@Edge는 AWSServiceRoleForLambdaReplicator 및 AWSServiceRoleForCloudFrontLogger이라는 서비스 연결 역할을 사용합니다. 다음 단원에서는 이러한 각 역할에 대한 권한을 설명합니다.

Lambda Replicator의 서비스 연결 역할 권한

This service-linked role allows Lambda to replicate Lambda@Edge functions to AWS Regions.

AWSServiceRoleForLambdaReplicator 서비스 연결 역할은 그 역할을 위임하기 위해 다음 서비스를 신뢰합니다.

- `replicator.lambda.amazonaws.com`

역할 권한 정책에서는 Lambda@Edge가 지정된 리소스에서 다음 작업을 완료할 수 있도록 허용합니다.

- 작업: `lambda:CreateFunction`에 대한 `arn:aws:lambda:*:*:function:*`
- 작업: `lambda:DeleteFunction`에 대한 `arn:aws:lambda:*:*:function:*`
- 작업: `lambda:DisableReplication`에 대한 `arn:aws:lambda:*:*:function:*`
- 작업: `iam:PassRole`에 대한 `all AWS resources`
- 작업: `cloudfront:ListDistributionsByLambdaFunction`에 대한 `all AWS resources`

CloudFront Logger에 대한 서비스 연결 역할 권한

이 서비스 연결 역할을 사용하면 Lambda@Edge 확인 오류를 디버깅하기 위해 CloudFront에서 CloudWatch 계정으로 로그 파일을 푸시할 수 있습니다.

AWSServiceRoleForCloudFrontLogger 서비스 연결 역할은 그 역할을 위임하기 위해 다음 서비스를 신뢰합니다.

- `logger.cloudfront.amazonaws.com`

역할 권한 정책에서는 Lambda@Edge가 지정된 리소스에서 다음 작업을 완료할 수 있도록 허용합니다.

- 작업: `logs:CreateLogGroup`에 대한 `arn:aws:logs:*:*:/aws/cloudfront/*`
- 작업: `logs:CreateLogStream`에 대한 `arn:aws:logs:*:*:/aws/cloudfront/*`
- 작업: `logs:PutLogsEvent`에 대한 `arn:aws:logs:*:*:/aws/cloudfront/*`

IAM 개체(예: 사용자, 그룹 또는 역할)가 Lambda@Edge 서비스 연결 역할을 삭제할 수 있도록 권한을 구성해야 합니다. 자세한 내용은 IAM 사용 설명서의 서비스 연결 역할 권한 단원을 참조하십시오.

Lambda@Edge의 서비스 연결 역할 생성

일반적으로 Lambda@Edge에 대한 서비스 연결 역할은 수동으로 생성하지 않습니다. 다음 시나리오에서 서비스는 역할을 자동으로 생성합니다.

- 트리거를 처음으로 생성할 때, 이 서비스에서는 AWSServiceRoleForLambdaReplicator(이)라는 역할이 없는 경우 해당 역할을 생성합니다. 이 역할은 Lambda에서 AWS 리전으로 Lambda@Edge 함수를 복제하도록 합니다.

  서비스 연결 역할을 삭제하는 경우, 배포에서 Lambda@Edge의 새 트리거를 추가할 때 이 역할이 다시 생성됩니다.
- Lambda@Edge 연결이 있는 CloudFront 배포를 업데이트 또는 생성할 때, 이 서비스에서는 AWSServiceRoleForCloudFrontLogger(이)라는 역할이 없는 경우 해당 역할을 생성합니다. 이 역할은 CloudFront가 CloudWatch로 로그 파일을 푸시하도록 합니다.

  서비스 연결 역할을 삭제했다 하더라도, Lambda@Edge 연결이 있는 CloudFront 배포를 업데이트 또는 생성하는 경우 해당 역할이 다시 생성됩니다.

이러한 서비스 연결 역할을 수동으로 만들어야 하는 경우 AWS CLI를 사용하여 다음 명령을 실행합니다.

AWSServiceRoleForLambdaReplicator 역할을 생성하려면

```
aws iam create-service-linked-role --aws-service-name
replicator.lambda.amazonaws.com
```

AWSServiceRoleForCloudFrontLogger 역할을 생성하려면

```
aws iam create-service-linked-role --aws-service-name
logger.cloudfront.amazonaws.com
```

Lambda@Edge 서비스 연결 역할 편집

Lambda@Edge에서는 AWSServiceRoleForLambdaReplicator 또는 AWSServiceRoleForCloudFrontLogger 서비스 연결 역할을 편집할 수 없습니다. 서비스에서 서비스 연결 역할을 만든 후에는 다양한 개체가 역할을 참조할 수 있기 때문에 역할 이름을 변경할 수 없습니다. 그러나 IAM를 사용하여 역할의 설명을 편집할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 서비스에 연결 역할 편집 단원을 참조하십시오.

Lambda@Edge 서비스 연결 역할 삭제

Lambda@Edge를 더 이상 사용할 필요가 없는 경우 해당 서비스 연결 역할을 삭제하는 것이 좋습니다. 따라서 적극적으로 모니터링하지 않거나 유지하지 않는 미사용 개체가 없도록 합니다. 단, 계정에서 Lambda@Edge 리소스를 먼저 정리해야 역할을 수동으로 삭제할 수 있습니다.

배포에서 모든 Lambda@Edge 연결을 제거하려면 배포를 업데이트하여 모든 Lambda@Edge 함수 트리거를 제거하거나, Lambda@Edge 함수를 사용하는 배포 자체를 제거하십시오. 자세한 내용은 Lambda@Edge 함수 및 복제본 삭제 (p. 315) 단원을 참조하십시오.

모든 Lambda@Edge 함수 연결을 배포에서 제거하고 CloudFront가 AWS 위치에서 함수 복제본을 제거하고 나면 서비스 연결 역할을 삭제할 수 있습니다.

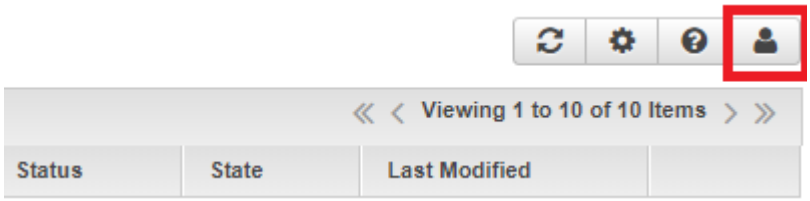
> **Note**
>
> CloudFront의 업데이트가 완료되지 않으면 서비스 연결 역할이 삭제되지 않습니다. 이 문제가 발생하면 몇 분 기다렸다가 삭제 단계를 다시 시도하십시오.

각 서비스 연결 역할을 수동으로 삭제하려면 별도의 절차를 따라야 합니다.

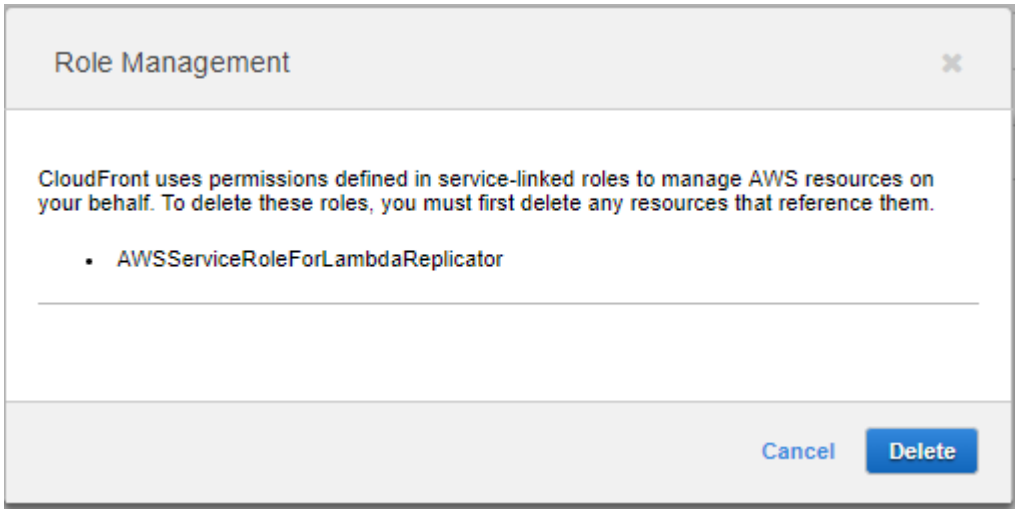
- CloudFront 콘솔을 사용하여 AWSServiceRoleForLambdaReplicator 역할을 삭제합니다.
- IAM 콘솔을 사용하여 AWSServiceRoleForCloudFrontLogger 역할을 삭제합니다.

수동으로 AWSServiceRoleForLambdaReplicator 서비스 연결 역할을 삭제하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. CloudFront 배포 페이지에서 상단 오른쪽에 있는 아바타를 클릭합니다.



3. 삭제를 선택합니다.



수동으로 AWSServiceRoleForCloudFrontLogger 서비스 연결 역할을 삭제하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/iam/에서 IAM 콘솔을 엽니다.
2. IAM 콘솔의 탐색 창에서 역할을 선택합니다. 그런 다음 삭제할 역할의 이름이나 행이 아닌 이름 옆에 있는 확인란을 선택합니다.
3. 페이지 상단의 역할 작업에서 역할 삭제를 선택합니다.
4. 확인 대화 상자가 나타나면 서비스 마지막 액세스 데이터를 검토합니다. 이 데이터는 선택한 각 역할이 AWS 서비스를 마지막으로 액세스한 일시를 보여 줍니다. 이를 통해 역할이 현재 활동 중인지 여부를 확인할 수 있습니다. 계속 진행하려면 [Yes, Delete]을 선택하여 삭제할 서비스 연결 역할을 제출합니다.
5. IAM 콘솔 알림을 보고 서비스 연결 역할 삭제 진행 상황을 모니터링합니다. IAM 서비스 연결 역할 삭제는 비동기이므로 삭제할 역할을 제출한 후에 삭제 작업이 성공하거나 실패할 수 있습니다. 자세한 정보는 IAM 사용 설명서의 서비스에 연결 역할 삭제 단원을 참조하십시오.

CloudFront 서비스 연결 역할에 대해 지원되는 리전

CloudFront은(는) 다음 리전에서 Lambda@Edge에 대한 서비스 연결 역할 사용을 지원합니다.

리전 이름	리전 자격 증명	CloudFront의 지원
미국 동부(버지니아 북부)	us-east-1	예

리전 이름	리전 자격 증명	CloudFront의 지원
미국 동부(오하이오)	us-east-2	예
미국 서부(오레곤)	us-west-2	예
아시아 태평양(뭄바이)	ap-south-1	예
아시아 태평양(도쿄)	ap-northeast-1	예
아시아 태평양(서울)	ap-northeast-2	예
아시아 태평양(싱가포르)	ap-southeast-1	예
아시아 태평양(시드니)	ap-southeast-2	예
유럽(프랑크푸르트)	eu-central-1	예
유럽(런던)	eu-west-2	예
남아메리카(상파울루)	sa-east-1	예

Lambda@Edge 함수 작성 및 생성

Lambda@Edge를 사용하려면 Lambda 함수에 대한 코드를 작성한 다음, 특정 CloudFront 이벤트(트리거)를 기반으로 함수를 실행하도록 AWS Lambda을 설정합니다. 함수를 실행하도록 Lambda를 설정하려면 Lambda에서 함수 생성 옵션을 사용합니다.

AWS 콘솔을 사용해 Lambda 함수 및 CloudFront 트리거를 처리하거나 API를 사용하여 프로그래밍 방식으로 Lambda@Edge를 처리할 수 있습니다.

- 콘솔을 사용하는 경우에는 Lambda 함수를 생성하는 데 AWS Lambda 콘솔만 사용할 수 있다는 점에 유의하십시오. Amazon CloudFront 콘솔을 사용해서는 함수를 생성할 수 없습니다.
- 프로그래밍 방식으로 Lambda@Edge를 처리하는 데 도움이 되는 몇 가지 리소스가 있습니다. 자세한 내용은 프로그래밍 방식으로 Lambda 함수 및 CloudFront 트리거 생성 (p. 301) 단원을 참조하십시오.

> **Note**
>
> Lambda@Edge 함수에 트리거를 추가하기 위해 AWS Lambda 콘솔 또는 CloudFront 콘솔을 사용할 수 있습니다.

주제

- Lambda@Edge의 함수 작성 (p. 298)
- Lambda 콘솔에서 Lambda@Edge 함수 생성 (p. 299)
- Lambda@Edge의 Lambda 함수 편집 (p. 300)
- 프로그래밍 방식으로 Lambda 함수 및 CloudFront 트리거 생성 (p. 301)

Lambda@Edge의 함수 작성

Lambda@Edge 함수 작성에 도움이 되는 몇 가지 리소스가 있습니다.

- Lambda@Edge에서 사용할 수 있는 Lambda 함수 작성에 대한 자세한 내용은 Lambda 함수에 대한 요구 사항 및 제한 사항 (p. 357) 단원을 참조하십시오.
- Lambda@Edge 함수에서 사용할 이벤트 구조에 대해 자세히 알아보려면 Lambda@Edge 이벤트 구조 (p. 315) 단원을 참조하십시오.

- A/B 테스트용 함수 및 HTTP 리디렉션 생성용 함수와 같은 Lambda@Edge 함수의 예제를 보려면 Lambda@Edge 함수 예제 (p. 330) 단원을 참조하십시오.

Lambda@Edge에서 Node.js 또는 Python을 사용하는 프로그래밍 모델은 AWS 리전에서 Lambda를 사용하는 것과 동일합니다. 자세한 내용은 Node.js를 사용하여 Lambda 함수 작성 또는 Python을 사용하여 Lambda 함수 작성을 참조하십시오.

Lambda@Edge 코드에 `callback` 파라미터를 포함시키고 요청 또는 응답 이벤트에 해당되는 객체를 반환합니다.

- 요청 이벤트 – 응답에 `cf.request` 객체를 포함합니다.

  응답을 생성하는 경우 응답에 `cf.response` 객체를 포함합니다. 자세한 내용은 요청 트리거에서 HTTP 응답 생성 (p. 326) 단원을 참조하십시오.
- 응답 이벤트 – 응답에 `cf.response` 객체를 포함합니다.

Lambda 콘솔에서 Lambda@Edge 함수 생성

CloudFront 이벤트를 기반으로 한 Lambda 함수를 실행하도록 AWS Lambda를 설정하려면 다음 절차를 수행합니다.

Lambda@Edge 함수를 생성하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/lambda/에서 AWS Lambda 콘솔을 엽니다.
2. 이미 하나 이상의 Lambda 함수가 있는 경우에는 함수 생성을 선택합니다.

   아무 함수도 없는 경우 Get Started Now(지금 시작하기)를 선택합니다.
3. 페이지 상단의 리전 목록에서 미국 동부(버지니아 북부)을 선택합니다.
4. 자체 코드를 사용하여 함수를 생성하거나 CloudFront 블루프린트로 시작하는 함수를 생성합니다.

   - 자체 코드를 사용하여 함수를 생성하려면 새로 작성을 선택합니다.
   - CloudFront용 블루프린트 목록을 표시하려면 필터 필드에 cloudfront를 입력한 다음, Enter 키를 누릅니다.

     사용할 블루프린트를 찾은 경우 블루프린트 이름을 선택합니다.
5. 기본 정보 섹션에서 다음 값을 지정합니다.

   이름

   함수 이름을 입력합니다.

   역할

   템플릿에서 새 역할 생성을 선택합니다.

   Note

   이 값을 선택하면 빠르게 시작할 수 있습니다. 또는 기존 역할 선택이나 사용자 지정 역할 생성을 선택할 수 있습니다. 다음 중 하나를 선택한 경우 프롬프트의 메시지를 따라 이 단원의 정보를 모두 입력합니다.

   역할 이름

   역할의 이름을 입력합니다.

   정책 템플릿

   기본 Edge Lambda 권한을 선택합니다.

6. 4단계에서 새로 작성을 선택한 경우 7단계로 건너뜁니다.

   4단계에서 블루프린트를 선택한 경우 cloudfront 섹션에서 이 함수를 CloudFront 배포 및 CloudFront 이벤트의 캐시와 연결하는 트리거를 하나 생성할 수 있습니다. 이때 함수 생성 시 트리거가 없도록 제거를 선택하는 것이 좋습니다. 그런 다음, 나중에 트리거를 추가할 수 있습니다.

   > **Important**
   >
   > 나중에 트리거를 추가하는 이유는 무엇일까요? 보통은 트리거를 추가하기 전에 함수를 테스트하고 디버깅하는 것이 가장 좋습니다. 대신 지금 트리거를 추가하면 함수가 생성되어 전 세계의 AWS 리전으로 복제를 완료하고 난 후 해당 배포가 배포되는 즉시 함수가 실행되기 시작합니다.

7. [Create function]을 선택합니다.

   Lambda는 함수의 두 버전인 $LATEST와 Version 1을 생성합니다. $LATEST 버전만 편집할 수 있지만 콘솔에 처음에는 Version 1이 표시됩니다.
8. 함수를 편집하려면 페이지 위쪽의, 함수 ARN 아래에서 Version 1을 선택합니다. 그런 다음, 버전 탭에서 $LATEST를 선택합니다. 함수에서 나간 후 다시 돌아오면 버튼 레이블이 한정자로 바뀌어 있습니다.
9. 구성 탭에서 해당하는 코드 입력 유형을 선택합니다. 그런 다음, 프롬프트의 메시지를 따라 코드를 편집하거나 업로드합니다.
10. 실행 시간에서 함수 코드를 기반으로 값을 선택합니다.
11. 태그 섹션에서 해당하는 태그를 추가합니다.
12. 작업을 선택한 다음, Publish new version(새 버전 게시)을 선택합니다.
13. 새 함수 버전의 설명을 입력합니다.
14. [Publish]를 선택합니다.
15. 함수를 테스트하고 디버깅합니다. Lambda 콘솔에서 테스트에 대한 자세한 내용은 AWS Lambda Developer Guide의 콘솔로 Lambda 함수 생성의 Lambda 함수를 호출하고 결과, 로그 및 지표 확인 단원을 참조하십시오.
16. CloudFront 이벤트에 대해 함수를 실행할 준비가 되면 다른 버전을 게시하고 함수를 편집하여 트리거를 추가합니다. 자세한 내용은 Lambda@Edge 함수에 대한 트리거 추가 (p. 302) 단원을 참조하십시오.

Lambda@Edge의 Lambda 함수 편집

Lambda 함수를 편집하려면 다음 사항에 유의하십시오.

- 원래 버전은 $LATEST라고 표시되어 있습니다.
- $LATEST 버전만 편집할 수 있습니다.
- $LATEST 버전을 편집할 때마다 번호가 지정된 새 버전을 게시해야 합니다.
- $LATEST에 대한 트리거는 생성할 수 없습니다.
- 새 버전의 함수를 게시하면 Lambda는 이전 버전의 트리거를 새 버전으로 자동 복사하지 않습니다. 새 버전에 대한 트리거를 다시 생성해야 합니다.
- CloudFront 이벤트에 대한 트리거를 함수에 추가할 때 동일한 함수의 이전 버전에 대해 동일한 배포, 캐시 동작 및 이벤트에 대한 트리거가 이미 존재하면 Lambda는 이전 버전에서 트리거를 삭제합니다.
- 트리거를 추가하는 등 CloudFront 배포를 업데이트한 후에는 변경 사항이 엣지 로케이션으로 전파될 때까지 기다려야만 트리거에서 지정한 함수가 작동합니다.

Lambda 함수를 편집하려면(AWS Lambda 콘솔)

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/lambda/에서 AWS Lambda 콘솔을 엽니다.
2. 페이지 상단의 리전 목록에서 미국 동부(버지니아 북부)을 선택합니다.
3. 함수 목록에서 편집하려는 함수의 이름을 선택합니다.

기본 설정 사용 시, $LATEST 버전이 콘솔에 표시됩니다. 이전 버전을 볼 수 있지만(한정자 선택) $LATEST만 편집할 수 있습니다.

4. 코드 탭의 코드 입력 항목에서 브라우저에서 코드를 편집하거나, .zip 파일을 업로드하거나, Amazon S3에서 파일을 업로드하도록 선택합니다.
5. 저장 또는 저장 및 테스트를 선택합니다.
6. 작업 및 Publish new version(새 버전 게시)을 차례대로 선택합니다.
7. $LATEST의 새 버전 게시 대화 상자에서 새 버전 설명을 입력합니다. 이 설명은 자동으로 생성된 버전 번호와 함께 버전 목록에 표시됩니다.
8. [Publish]를 선택합니다.

   새 버전은 자동으로 최신 버전이 됩니다. 버전 번호는 페이지의 왼쪽 위 모서리의 버전 버튼에 표시됩니다.
9. 트리거 탭을 선택합니다.
10. 트리거 추가를 선택합니다.
11. 트리거 추가 대화 상자에서 점이 있는 상자를 선택한 다음, CloudFront를 선택합니다.

    Note

    함수에 대해 하나 이상의 트리거를 이미 만든 경우 CloudFront가 기본 서비스입니다.
12. 다음 값을 지정하여 Lambda 함수를 실행할 시기를 나타냅니다.

    배포 ID

    트리거를 추가할 배포의 ID를 선택합니다.

    캐시 동작

    함수를 실행할 객체를 지정하는 캐시 동작을 선택합니다.

    CloudFront 이벤트

    함수를 실행시키는 CloudFront 이벤트를 선택합니다.

    트리거 활성화 및 복제

    이 확인란을 선택하면 Lambda가 전 세계 리전으로 함수를 복제합니다.
13. [Submit]을 선택합니다.
14. 이 함수에 대한 트리거를 추가하려면 10~13단계를 반복합니다.

프로그래밍 방식으로 Lambda 함수 및 CloudFront 트리거 생성

AWS 콘솔을 사용하는 대신 API 작업을 사용해 Lambda@Edge 함수 및 CloudFront 트리거를 프로그래밍 방식으로 설정할 수 있습니다. 자세한 정보는 다음 단원을 참조하십시오.

- AWS Lambda Developer Guide의 API 참조
- Amazon CloudFront API Reference
- AWS CLI
  - Lambda create-function 명령
  - CloudFront create-distribution 명령
  - CloudFront create-distribution-with-tags 명령
  - CloudFront update-distribution 명령
- AWS SDK(SDK 및 도구 키트 단원 참조)

- PowerShell용 AWS 도구 Cmdlet Reference

Lambda@Edge 함수에 대한 트리거 추가

Lambda@Edge 트리거는 CloudFront 배포, 캐시 동작, 그리고 함수 실행을 유도하는 이벤트를 하나로 조합한 것입니다. 함수를 실행시키는 CloudFront 트리거를 하나 이상 지정할 수 있습니다. 예를 들어, 최종 사용자가 해당 배포에 설정된 특정 캐시 동작을 CloudFront에 요청하면 함수가 실행되도록 하는 트리거를 생성할 수 있습니다.

> **Tip**
>
> 다음은 CloudFront 캐시 동작에 익숙하지 않은 사용자를 위한 간단한 개요입니다. CloudFront 배포를 생성할 때 다른 요청을 수신할 때 응답하는 방법을 CloudFront에게 알려주는 설정을 지정합니다. 기본 설정을 배포에 대한 기본 캐시 동작이라고 합니다. CloudFront가 특정한 상황(예를 들어, 특정 파일 유형에 대한 요청을 수신할 때)에서 응답하는 방법을 정의하는 추가 캐시 동작을 설정할 수 있습니다. 자세한 내용은 캐시 동작 설정 단원을 참조하십시오.

Lambda 함수를 생성할 때는 트리거를 하나만 지정할 수 있습니다. 하지만 나중에 Lambda 콘솔을 사용하거나 CloudFront 콘솔에서 배포를 편집하여 동일한 함수에 더 많은 트리거를 추가할 수 있습니다.

- Lambda 콘솔을 사용하는 방법은 동일한 CloudFront 배포에 대한 함수에 트리거를 더 추가하려는 경우에 적합합니다.
- CloudFront 콘솔을 사용하는 방법은 업데이트하려는 배포를 쉽게 찾을 수 있기 때문에 여러 배포에 대한 트리거를 추가하려는 경우에 더 적합할 수 있습니다. 또한 동시에 다른 CloudFront 설정을 업데이트할 수 있습니다.

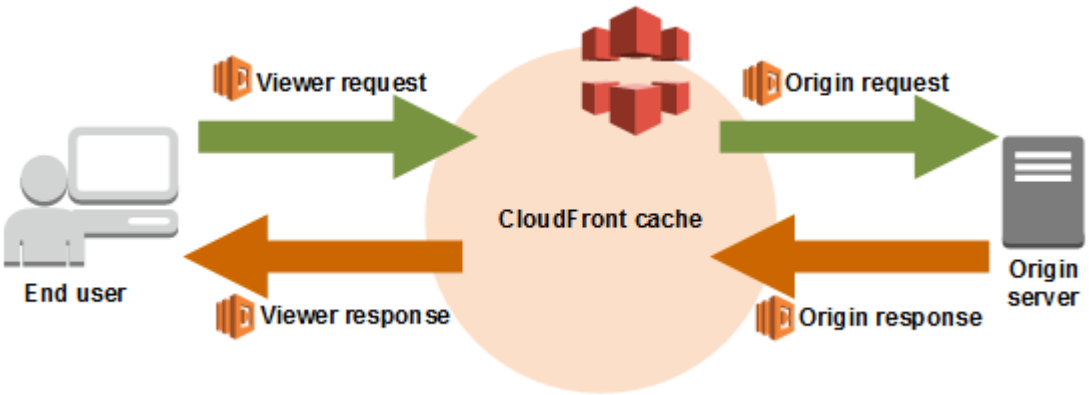
> **Note**
>
> 프로그래밍 방식으로 Lambda@Edge를 처리하는 데 도움이 되는 몇 가지 리소스가 있습니다. 자세한 내용은 프로그래밍 방식으로 Lambda 함수 및 CloudFront 트리거 생성 (p. 301) 단원을 참조하십시오.

주제

- Lambda 함수를 트리거할 수 있는 CloudFront 이벤트 (p. 302)
- Lambda 함수를 트리거하는 데 사용할 CloudFront 이벤트를 결정하는 방법 (p. 304)
- Lambda 콘솔을 사용하여 트리거 추가 (p. 304)
- CloudFront 콘솔을 사용하여 트리거 추가 (p. 305)

Lambda 함수를 트리거할 수 있는 CloudFront 이벤트

CloudFront 배포 내 각 캐시 동작에 대해 다음과 같은 CloudFront 이벤트가 발생할 경우 Lambda함수가 실행되도록 하는 트리거(연결)를 최대 4개까지 추가할 수 있습니다. CloudFront 트리거는 아래 그림의 네 가지 CloudFront 이벤트 중 하나를 기반으로 실행됩니다.



Lambda@Edge 함수를 트리거하는 데 사용할 수 있는 CloudFront 이벤트는 다음과 같습니다.

최종 사용자 요청

최종 사용자로부터 요청을 받은 CloudFront가 요청된 객체가 CloudFront 캐시에 있는지 확인하고 나면 함수가 실행됩니다.

오리진 요청

오직 CloudFront가 오리진으로 요청을 전달할 때만 함수가 실행됩니다. 요청된 객체가 CloudFront 캐시에 저장되어 있으면 함수는 실행되지 않습니다.

오리진 응답

CloudFront가 오리진으로부터 응답을 수신한 후 그리고 응답에 객체를 캐싱하기 전에 함수가 실행됩니다. 오리진에서 오류가 반환되는 경우에도 함수는 실행됩니다.

다음과 같은 경우에는 함수가 실행되지 않습니다.

- 요청된 파일이 CloudFront 캐시에 있고 만료되지 않은 경우.
- 오리진 요청 이벤트가 트리거한 함수로부터 응답이 생성되는 경우.

최종 사용자 응답

요청된 파일을 최종 사용자에게 반환하기 전에 함수가 실행됩니다. 이때 함수는 해당 파일이 이미 CloudFront 캐시에 있는지 여부와 상관없이 실행됩니다.

다음과 같은 경우에는 함수가 실행되지 않습니다.

- 오리진이 400 이상의 HTTP 상태 코드 반환하는 경우
- 사용자 지정 오류 페이지가 반환될 때.
- 최종 사용자 요청 이벤트가 트리거한 함수로부터 응답이 생성되는 경우.
- CloudFront가 자동으로 HTTP 요청을 HTTPS로 리디렉션하는 경우(Viewer Protocol Policy (p. 45)의 값이 Redirect HTTP to HTTPS(HTTP를 HTTPS로 재지정)일 경우)

한 캐시 동작에 여러 트리거를 추가하는 경우, 이를 사용하여 동일한 함수를 실행하거나 트리거마다 다른 함수를 실행할 수 있습니다. 또한 복수의 배포에 동일한 함수를 연결할 수도 있습니다.

Note

CloudFront 이벤트가 Lambda 함수 실행을 트리거하면 해당 함수가 완료되어야 CloudFront가 계속 작동할 수 있습니다. 예를 들어, CloudFront 최종 사용자 요청 이벤트에 따라 Lambda 함수가 트리거되면 CloudFront는 Lambda 함수 실행이 완료될 때까지 최종 사용자에게 응답을 반환하거나 오리진에 요청을 전달하지 않습니다. 다시 말해, Lambda 함수를 트리거하는 각 요청은 요청 지연 시간을 증가시키므로 최대한 빨리 함수가 실행되도록 해야 합니다.

Lambda 함수를 트리거하는 데 사용할 CloudFront 이벤트를 결정하는 방법

Lambda 함수를 트리거하는 데 사용할 CloudFront 이벤트를 결정할 때 다음 사항을 고려해야 합니다.

CloudFront에서 Lambda 함수가 변경한 객체를 캐싱하는지 여부

CloudFront가 Lambda 함수로 수정된 객체를 캐싱하여 다음에 해당 객체가 요청될 경우 엣지 로케이션의 객체에 CloudFront가 서비스를 제공할 수 있게 하려면 오리진 요청 또는 오리진 응답 이벤트를 사용합니다. 그러면 오리진에 대한 로드가 감소하고, 이후 요청의 지연 시간이 단축되며, 이후 요청에서 Lambda@Edge를 호출하는 비용이 절감됩니다.

예를 들어, 오리진에 의해 반환되는 객체의 헤더를 추가, 제거 또는 변경하고 CloudFront에서 해당 결과를 캐싱하게 하려는 경우 오리진 응답 이벤트를 사용합니다.

모든 요청에 대해 함수를 실행할지 여부

CloudFront가 배포에 대해 수신하는 모든 요청에서 함수를 실행하려는 경우 최종 사용자 요청 또는 최종 사용자 응답 이벤트를 사용합니다. 오리진 요청 및 오리진 응답 이벤트는 요청된 객체가 엣지 로케이션에 캐싱되지 않고 CloudFront가 오리진으로 요청을 전달하는 경우에만 발생합니다.

함수가 캐시 키를 변경하는지 여부

함수가 캐싱의 기준으로 사용되는 값을 변경하게 하려는 경우 최종 사용자 요청 이벤트를 사용합니다. 예를 들어, 함수가 URL을 변경하여 경로에 언어 약자를 포함시키는 경우(예를 들어, 사용자가 드롭다운 목록에서 언어를 선택했기 때문) 최종 사용자 요청 이벤트를 사용합니다.

- 최종 사용자 요청 내 URL – http://example.com/en/index.html
- 요청이 독일 내 IP 주소로부터 온 경우의 URL – http://example.com/de/index.html

쿠키 또는 요청 헤더를 기준으로 캐싱하는 경우에도 최종 사용자 요청 이벤트를 사용합니다.

Note

함수가 쿠키 또는 헤더를 변경하는 경우 CloudFront가 요청의 해당 부분을 오리진으로 전달하도록 구성합니다. 자세한 내용은 다음 주제를 참조하십시오.

- 쿠키를 기반으로 콘텐츠 캐싱 (p. 198)
- 요청 헤더에 따라 콘텐츠 캐싱 (p. 200)

함수가 오리진으로부터의 응답에 영향을 미치는지 여부

함수가 오리진으로부터의 응답에 영향을 미치는 방식으로 요청을 변경하게 하려는 경우 오리진 요청 이벤트를 사용합니다. 일반적으로 대부분의 최종 사용자 요청 이벤트는 오리진으로 전달되지 않습니다. CloudFront는 이미 엣지 캐시에 저장된 객체를 사용하여 요청에 응답합니다. 함수가 오리진 요청 이벤트를 기준으로 요청을 변경하는 경우 CloudFront가 변경된 오리진 요청에 대한 응답을 캐싱합니다.

Lambda 콘솔을 사용하여 트리거 추가

Lambda@Edge 함수(AWS Lambda 콘솔)에 트리거를 추가하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/lambda/에서 AWS Lambda 콘솔을 엽니다.
2. 페이지 상단의 리전 목록에서 미국 동부(버지니아 북부)을 선택합니다.
3. 함수 페이지에서 트리거를 추가할 함수의 이름을 선택합니다.
4. 한정자를 선택한 다음 버전 탭을 선택합니다.
5. 트리거를 추가할 버전을 선택합니다.

Important

$LATEST 버전에 대한 트리거를 만들 수 없으므로 번호가 지정된 버전에 대해 트리거를 만들어야 합니다.

버전을 선택한 후 버튼의 이름이 버전: $LATEST 또는 버전: 버전 번호로 변경됩니다.

6. 트리거 탭을 선택합니다.
7. 트리거 추가를 선택합니다.
8. 트리거 추가 대화 상자에서 점이 있는 상자를 선택한 다음, CloudFront를 선택합니다.

   Note

   하나 이상의 트리거를 이미 만든 경우 CloudFront가 기본 서비스입니다.
9. 다음 값을 지정하여 Lambda 함수를 실행할 시기를 나타냅니다.

   배포 ID

   트리거를 추가할 배포의 ID를 선택합니다.

   캐시 동작

   함수를 실행할 객체를 지정하는 캐시 동작을 선택합니다.

   Note

   캐시 동작에 *를 지정하는 경우 Lambda 함수는 기본 캐시 동작으로 배포됩니다.

   CloudFront 이벤트

   함수를 실행시키는 CloudFront 이벤트를 선택합니다.

   본문 포함

   함수의 요청 본문에 액세스하려는 경우 이 확인란을 선택합니다.

   트리거 활성화 및 복제

   이 확인란을 선택하면 AWS Lambda가 전 세계 리전으로 함수를 복제합니다.
10. [Submit]을 선택합니다.

    업데이트된 CloudFront 배포가 배포되면 함수가 지정된 CloudFront 이벤트에 대한 요청을 처리하기 시작합니다. 배포가 배포되었는지 확인하려면 탐색 창에서 배포를 선택합니다. 배포가 배포되면 배포의 상태 열에서 값이 진행 중에서 배포 완료로 변경됩니다.

CloudFront 콘솔을 사용하여 트리거 추가

Lambda 함수에 CloudFront 이벤트에 대한 트리거 추가(CloudFront 콘솔)

1. 트리거를 추가하려는 Lambda 함수의 ARN을 얻습니다.

   a. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/lambda/에서 AWS Lambda 콘솔을 엽니다.
   b. 페이지 상단의 리전 목록에서 미국 동부(버지니아 북부)을 선택합니다.
   c. 함수 목록에서 트리거를 추가하려는 함수의 이름을 선택합니다.
   d. 한정자를 선택하고 버전 탭을 선택한 다음, 트리거를 추가하려는 번호가 지정된 버전을 선택합니다.

      Important

      트리거는 $LATEST가 아니라 번호가 지정된 버전에만 추가할 수 있습니다.

e. 페이지 상단에 표시되는 ARN을 복사합니다. 예:

```
arn:aws:lambda:us-east-1:123456789012:function:TestFunction:2
```

끝부분의 숫자(이 예제에서는 2)가 함수의 버전 번호입니다.

2. https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
3. 배포 목록에서 트리거를 추가하려는 배포의 ID를 선택합니다.
4. 동작 탭을 선택합니다.
5. 트리거를 추가하려는 캐시 동작의 확인란을 선택한 다음 편집을 클릭합니다.
6. 이벤트 유형 목록의 Lambda Function Associations(Lambda 함수 연결)에서 함수를 실행하려는 시점, 즉 최종 사용자 요청, 최종 사용자 응답, 오리진 요청 또는 오리진 응답을 선택합니다.

   자세한 내용은 Lambda 함수를 트리거하는 데 사용할 CloudFront 이벤트를 결정하는 방법 (p. 304) 단원을 참조하십시오.
7. 선택한 이벤트가 발생하면 실행할 Lambda 함수의 ARN을 붙여 넣습니다. 이것은 1단계에서 복사한 값입니다.
8. 함수의 요청 본문에 액세스하려는 경우 본문 포함을 선택합니다.

   요청 본문을 바꾸기만 하려는 경우에는 이 옵션을 선택할 필요가 없습니다.
9. 더 많은 이벤트 유형에 대해 동일한 함수를 실행하려면 [+]를 선택한 다음 6 및 7단계를 반복합니다.
10. 예, 편집합니다를 선택합니다.
11. 이 배포에 대한 다른 캐시 동작에 트리거를 추가하려면 5~9단계를 반복합니다.

    업데이트된 CloudFront 배포가 배포되면 함수가 지정된 CloudFront 이벤트에 대한 요청을 처리하기 시작합니다. 배포가 배포되었는지 확인하려면 탐색 창에서 배포를 선택합니다. 배포가 배포되면 배포의 상태 열에서 값이 진행 중에서 배포 완료로 변경됩니다.

Lambda@Edge 함수 테스트 및 디버깅

이 주제에는 Lambda@Edge 함수 테스트 및 디버깅에 대한 전략을 설명하는 단원이 있습니다. 의도한 작업을 완료하는지 확인하기 위해서는 Lambda@Edge 함수 코드를 단독으로 테스트해야 하고, 함수가 CloudFront와 함께 잘 작동하는지 확인하기 위해서는 통합 테스트를 수행해야 합니다.

통합 테스트 중 또는 함수 배포 완료 후 HTTP 5xx 오류 등과 같은 CloudFront 오류를 디버깅해야 할 수 있습니다. Lambda 함수에서 반환되는 잘못된 응답, 함수 실행 시 발생하는 실행 오류 또는 Lambda 서비스의 실행 병목 현상 인한 오류 등이 있을 수 있습니다. 이 주제의 단원에서는 이러한 문제가 어떤 유형의 실패인지 확인한 다음 각 문제를 해결하기 위해 수행할 수 있는 단계에 대해 설명합니다.

> **Note**
>
> 문제를 해결하는 경우 CloudWatch 로그 파일 또는 지표를 검토할 때 함수가 실행된 위치에 가장 가까운 리전에서 로그 파일 또는 지표가 표시 또는 저장되는지 확인합니다. 예를 들어, 영국 내 사용자가 있는 웹사이트 또는 웹 애플리케이션이 있고 배포와 연결된 Lambda 함수가 있는 경우 런던 AWS 리전에 대한 CloudWatch 지표 또는 로그 파일을 볼 수 있도록 이 리전을 변경해야 합니다. 자세한 내용은 이 주제 마지막 부분에 있는 Lambda@Edge 리전 확인을 참조하십시오.

주제

- Lambda@Edge 함수 테스트 (p. 307)
- CloudFront에서 Lambda 함수 오류 파악 (p. 307)
- 잘못된 Lambda 함수 응답(확인 오류) 문제 해결하기 (p. 312)
- Lambda 함수 실행 오류 문제 해결 (p. 313)
- Lambda@Edge 리전 확인 (p. 313)
- 계정이 CloudWatch로 로그를 푸시하는지 확인 (p. 313)

Lambda@Edge 함수 테스트

Lambda 함수 테스트에는 독립 실행형 테스트와 통합 테스트, 이렇게 두 가지 단계가 있습니다.

독립 실행형 기능 테스트

CloudFront에 Lambda 함수를 추가하기 전에 Lambda 콘솔의 테스트 기능을 사용하거나 다른 방법을 사용하여 기능을 먼저 테스트해야 합니다. Lambda 콘솔에서 테스트에 대한 자세한 내용은 AWS Lambda Developer Guide의 콘솔로 Lambda 함수 생성의 Lambda 함수를 호출하고 결과, 로그 및 지표 확인 단원을 참조하십시오.

CloudFront에서 함수의 작동 상태를 보기 위한 테스트

함수가 배포와 연결되어 있고 CloudFront 이벤트를 기반으로 실행되는 경우에는 통합 테스트를 반드시 완료해야 합니다. 함수가 올바른 이벤트 발생 시 실행되고 CloudFront에 대해 올바른 응답을 반환하는지 확인합니다. 예를 들어, 이벤트 구조가 올바른지, 유효한 헤더만 포함되어 있는지 등을 확인합니다.

Lambda 콘솔에서 함수의 통합 테스트를 반복하는 것처럼 코드를 수정하거나 함수를 호출하는 CloudFront 트리거를 변경할 때 Lambda@Edge 자습서의 단계를 참조하십시오. 예를 들어, 본 자습서의 단계(4단계: 함수를 실행할 CloudFront 트리거 추가 (p. 288))에서 설명하는 것처럼 다양한 버전의 함수에서 작동하는지 확인합니다.

함수를 변경하고 배포할 때 업데이트된 함수 및 CloudFront 트리거가 모든 리전에서 복제되는데 몇 분 정도 걸릴 수 있습니다. 일반적으로 몇 분이면 되지만 최대 15분까지 걸릴 수 있습니다.

다음과 같이 CloudFront 콘솔에서 해당 배포를 보고 복제가 완료되었는지 확인할 수 있습니다.

- https://console.aws.amazon.com/cloudfront에서 CloudFront 콘솔로 이동합니다.

배포 상태가 진행 중에서 다시 배포 완료로 바뀌었는지 확인합니다. 이것은 함수가 복제되었다는 의미입니다. 이제 다음 단원의 단계에 따라 함수가 작동하는지 확인합니다.

콘솔의 테스트는 함수의 로직만 확인하며, Lambda@Edge에 고유한 모든 서비스 할당량(이전에는 제한이라고 함)은 적용하지 않습니다.

CloudFront에서 Lambda 함수 오류 파악

함수 로직이 올바르게 작동하는지 확인한 후 CloudFront에서 함수 실행 시 HTTP 5xx 오류가 나타날 수 있습니다. HTTP 5xx 오류는 여러 가지 이유로 반환될 수 있습니다. 여기에는 Lambda 함수 오류 또는 CloudFront의 다른 문제가 포함될 수 있습니다.

- Lambda@Edge 함수를 사용하는 경우 CloudFront 콘솔의 그래프를 사용하여 오류의 원인을 추적하고 문제를 해결할 수 있습니다. 예를 들어, HTTP 5xx 오류가 CloudFront 또는 Lambda 함수에 의해 발생한 것인지를 확인한 다음 특정 함수에 대해 관련 로그 파일을 검토하여 문제를 조사할 수 있습니다.
- CloudFront에서 일반적인 HTTP 오류 문제를 해결하려면 오리진의 오류 응답 문제 해결 (p. 221) 주제의 문제 해결 단계를 참조하십시오.

CloudFront에서 Lambda 함수 오류를 일으키는 원인

Lambda 함수가 HTTP 5xx 오류를 일으키는 원인에는 여러 가지가 있으며, 수행해야 할 문제 해결 단계는 오류 유형에 따라 달라집니다. 오류는 다음과 같이 분류할 수 있습니다.

Lambda 함수 실행 오류

함수에 처리되지 않은 예외가 있거나 코드에 오류가 있기 때문에 CloudFront가 Lambda에서 응답을 받지 못한 경우 실행 오류가 발생합니다. 예: 코드에 콜백이 포함된 경우(오류) 자세한 내용은 AWS Lambda Developer Guide의 Lambda 함수 오류 단원을 참조하십시오.

잘못된 Lambda 함수 응답이 CloudFront로 반환됨

함수 실행 후 CloudFront가 Lambda에서 응답을 수신했습니다. 응답의 객체 구조가 Lambda@Edge 이벤트 구조 (p. 315)를 따르지 않거나 응답에 잘못된 헤더 또는 기타 잘못된 필드가 포함되어 있는 경우 오류가 반환됩니다.

Lambda 서비스 할당량(이전에는 제한이라고 함)으로 인해 CloudFront에서의 실행이 제한됨

Lambda 서비스는 각 리전에서 실행을 조절하는데 할당량을 초과하면 오류를 반환합니다.

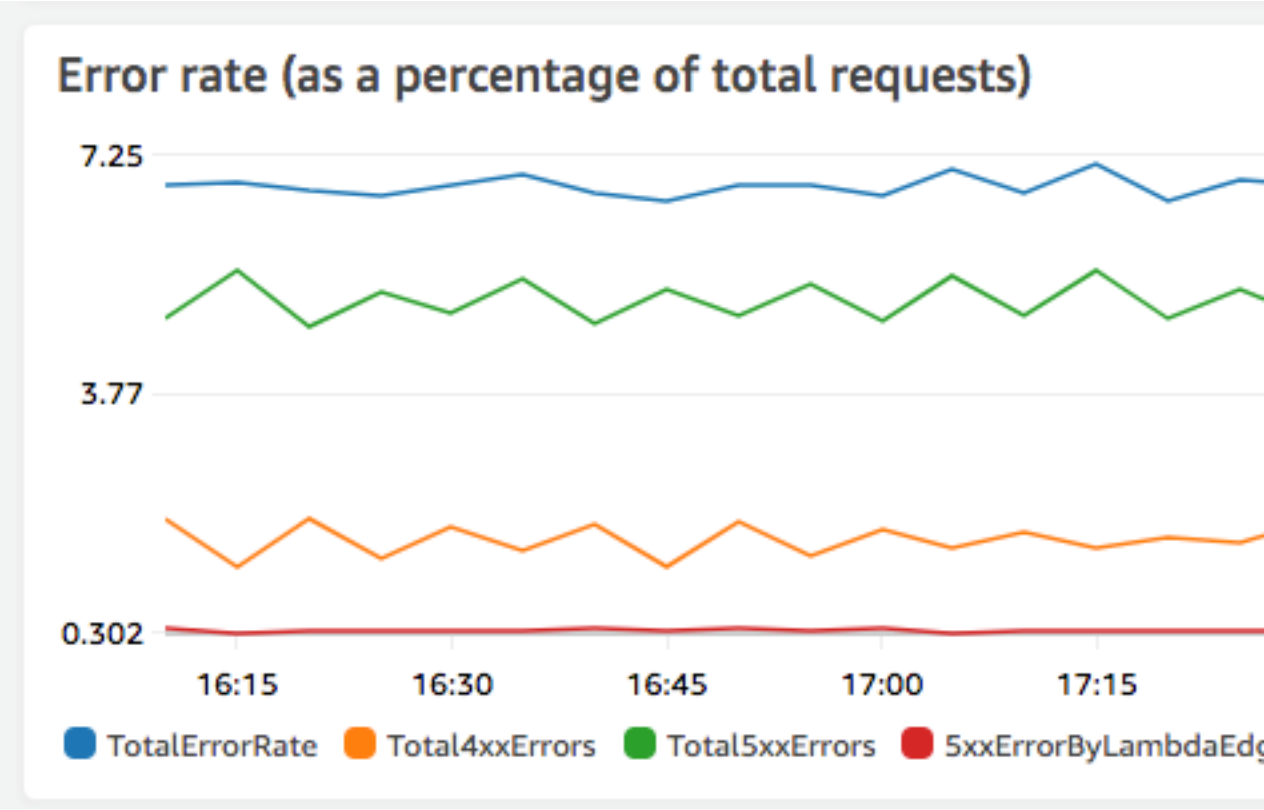
실패 유형을 확인하는 방법

디버깅할 때 집중할 위치를 결정하고 CloudFront에서 반환한 오류를 해결하려면 CloudFront에서 HTTP 오류가 반환된 이유를 파악하는 것이 좋습니다. 시작하려면 AWS Management 콘솔의 CloudFront 콘솔에 있는 모니터링 섹션에 제공된 그래프를 사용합니다. CloudFront 콘솔의 모니터링 섹션에서 그래프를 보는 방법에 대한 자세한 내용은 Amazon CloudWatch를 사용하여 CloudFront 모니터링 (p. 416) 단원을 참조하십시오.

다음 그래프는 오리진 또는 Lambda 함수로 오류를 반환하는지 여부를 추적하고 Lambda 함수의 오류일 경우 문제 유형의 범위를 좁히려 할 때 특히 유용합니다.

오류 발생율 그래프

각 배포의 개요 탭에서 볼 수 있는 그래프 중에는 오류 발생율 그래프가 있습니다. 이 그래프는 배포에 들어오는 총 요청의 백분율로 오류 발생율을 표시합니다. 그래프는 총 오류 발생율, 총 4xx 오류, 총 5xx 오류 및 Lambda 함수의 총 5xx 오류를 보여줍니다. 오류 유형 및 볼륨에 따라 원인을 조사하고 문제를 해결할 수 있는 단계를 수행할 수 있습니다.

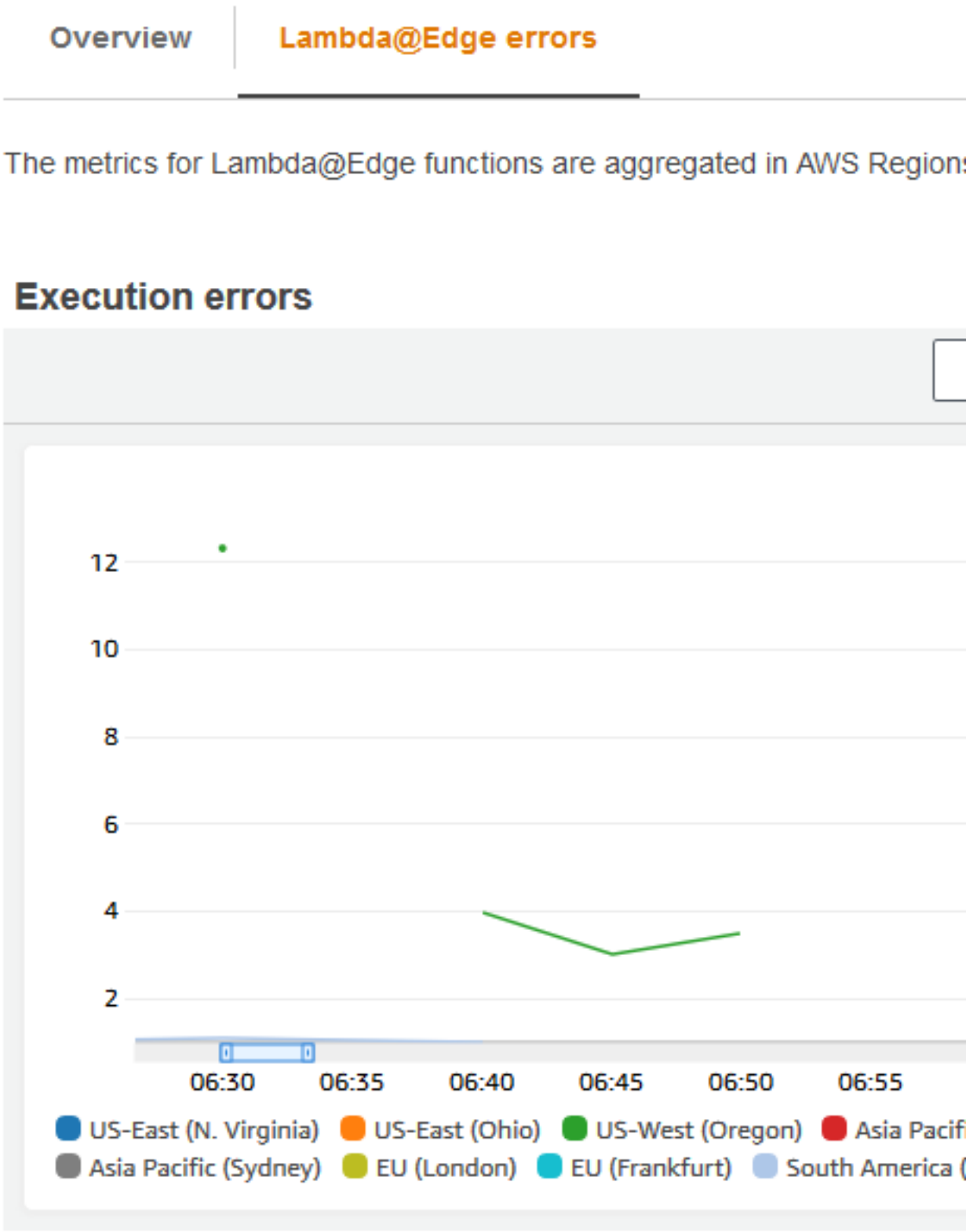


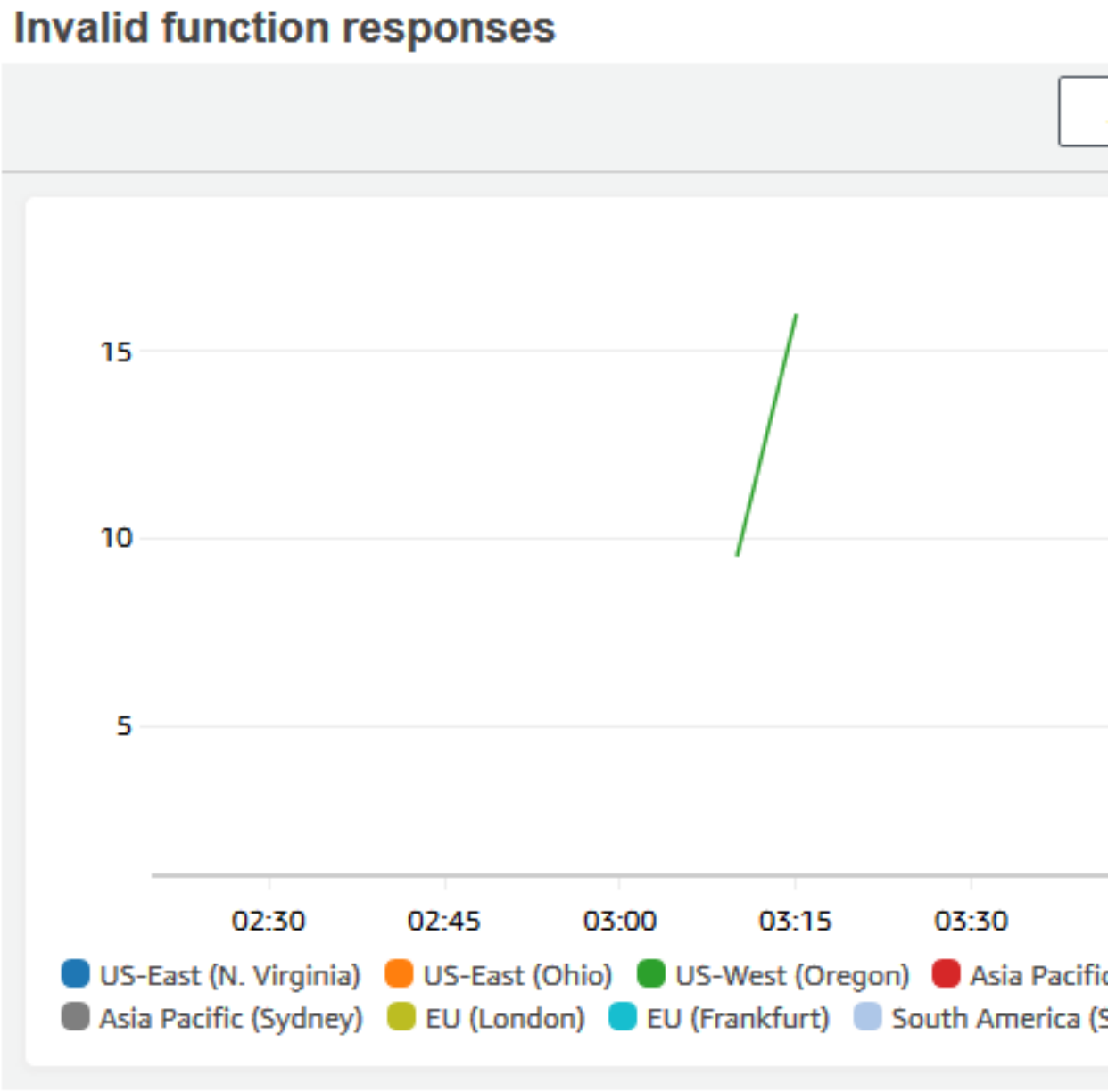
- Lambda 오류가 표시되면 함수가 반환하는 특정 유형의 오류를 확인하여 추가 조사를 수행할 수 있습니다. Lambda@Edge errors(Lambda@Edge 오류) 탭에는 함수 오류를 유형별로 분류하여 특정 함수에 대한 문제를 찾아낼 수 있는 그래프가 포함되어 있습니다.

- CloudFront 오류가 표시되면 문제를 해결하고 오리진 오류를 수정하거나 CloudFront 구성을 변경할 수 있습니다. 자세한 내용은 오리진의 오류 응답 문제 해결 (p. 221) 단원을 참조하십시오.

실행 오류 및 잘못된 함수 응답 그래프

Lambda@Edge errors(Lambda@Edge 오류) 탭에는 특정 배포에 대한 Lambda@Edge 오류를 유형별로 분류하는 그래프가 포함되어 있습니다. 예를 들어, 한 그래프는 모든 실행 오류를 AWS 리전별로 표시합니다. 동일한 페이지에서 문제를 쉽게 해결할 수 있도록 특정 함수에 대한 로그 파일을 열고 리전별로 검토하여 특정 문제를 찾을 수 있습니다. View execution error logs(실행 오류 로그 보기) 또는 View invalid function response(잘못된 함수 응답 보기) 로그에서 리전(및 실행 오류의 경우 함수)을 선택한 다음 로그 보기를 선택합니다.

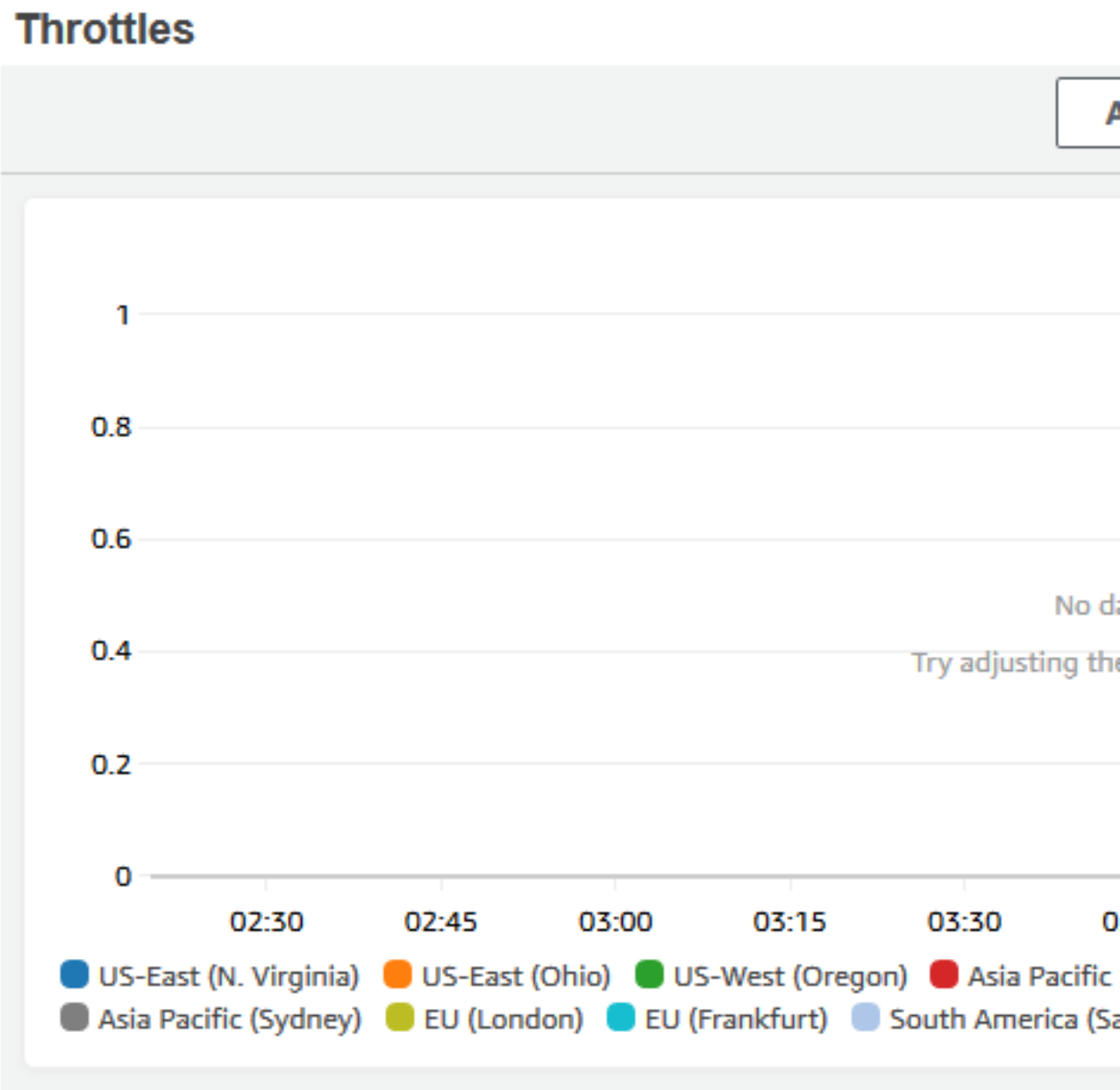

Overview
Lambda@Edge errors
The metrics for Lambda@Edge functions are aggregated in AWS Regions
Execution errors
12
10
8
6
4
2
06:30
06:35
06:40
06:45
06:50
06:55
US-East (N. Virginia)
US-East (Ohio)
US-West (Oregon)
Asia Pacif
Asia Pacific (Sydney)
EU (London)
EU (Frankfurt)
South America (



또한 이 장의 다음 단원에서 오류 해결 및 수정에 대한 추가 권장 사항을 읽으십시오.

제한 그래프

Lambda@Edge errors(Lambda@Edge 오류) 탭에는 제한 그래프도 있습니다. 경우에 따라 리전 동시성 할당량(이전에는 제한이라고 함)에 도달하면 Lambda 서비스가 리전별로 함수 호출을 제한합니다. 제한 초과 오류가 발생한 경우는 리전에서의 실행에 대해 Lambda 서비스가 부과한 할당량에 도달한 것입니다. 이러한 할당량을 늘리기 위한 요청 방법을 비롯한 자세한 내용은 Lambda@Edge에 대한 할당량 (p. 450) 단원을 참조하십시오.



이 정보를 사용하여 HTTP 오류 문제를 해결하는 방법에 대한 예제는 AWS에서 콘텐츠 전송을 디버깅하기 위한 네 가지 단계를 참조하십시오.

잘못된 Lambda 함수 응답(확인 오류) 문제 해결하기

Lambda 확인 오류가 문제라고 파악한 경우는 Lambda 함수가 CloudFront에 잘못된 응답을 반환하고 있다는 의미입니다. 이 단원의 지침에 따라 함수를 검토하는 단계를 수행한 다음 응답이 CloudFront 요구 사항을 따르는지 확인하십시오.

CloudFront에서는 다음과 같은 두 가지 방법으로 Lambda 함수의 응답을 확인합니다.

- Lambda 응답은 필수 객체 구조를 따라야 합니다. 잘못된 객체 구조의 예에는 구문 분석이 불가능한 JSON, 필수 필드 누락 및 응답에 포함된 잘못된 객체 등이 있습니다. 자세한 내용은 Lambda@Edge 이벤트 구조 (p. 315) 단원을 참조하십시오.
- 응답에는 올바른 객체만 포함되어 있어야 합니다. 응답에 올바른 객체가 포함되어 있는데 그 값이 지원되지 않는 경우에는 오류가 발생합니다. 블랙리스트에 등록된 헤더 또는 읽기 전용 헤더를 추가 또는 업데이트하는 경우(Lambda 함수에 대한 요구 사항 및 제한 사항 주제의 헤더 (p. 358) 단원 참조), 최대 본문 크기 초과(Lambda 응답 오류 (p. 328) 주제의 생성된 응답의 크기 제한 참조) 및 잘못된 문자 또는 값(Lambda@Edge 이벤트 구조 (p. 315) 단원 참조) 등을 예로 들 수 있습니다.

Lambda가 CloudFront에 잘못된 응답을 반환하면 로그 파일에 오류 메시지가 기록되고, CloudFront는 Lambda 함수가 실행된 리전의 CloudWatch로 로그 파일을 푸시합니다. 잘못된 응답이 있는 경우 CloudWatch로 로그 파일을 보내는 것은 기본 동작입니다. 그러나 Lambda 함수를 릴리스하기 전에 CloudFront와 연결한 경우에는 함수에 대해 활성화되지 않을 수 있습니다. 자세한 내용은 이 주제 뒷 부분에 나오는 계정이 CloudWatch로 로그를 푸시하는지 확인을 참조하십시오.

CloudFront는 배포와 연결된 로그 그룹 내에서 함수가 실행되는 위치에 해당하는 리전으로 로그 파일을 푸시합니다. 로그 그룹의 형식은 `/aws/cloudfront/LambdaEdge/`*`DistributionId`*인데, 여기서 *`DistributionId`*는 배포 ID입니다. CloudWatch 로그 파일을 찾을 수 있는 리전을 확인하려면 이 주제 뒷 부분에 나오는 Lambda@Edge 리전 확인을 참조하십시오.

오류를 재현할 수 있는 경우 오류가 발생하는 새로운 요청을 생성한 다음 실패한 CloudFront 응답(`X-Amz-Cf-Id` 헤더)에서 해당 요청 ID를 찾아 로그 파일에서 단일 실패를 확인할 수 있습니다. 로그 파일 항목에는 오류가 반환된 이유를 파악하는 데 도움이 되는 정보가 포함되어 있고 해당하는 Lambda 요청 ID가 나열되어 있기 때문에 단일 요청의 컨텍스트 내에서 근본 원인을 분석할 수 있습니다.

오류가 간헐적으로 발생하는 경우에는 CloudFront 액세스 로그를 사용하여 실패한 요청의 요청 ID를 찾은 다음 CloudWatch 로그에서 해당하는 오류 메시지를 검색할 수 있습니다. 자세한 내용은 앞의 실패 유형 확인 단원을 참조하십시오.

Lambda 함수 실행 오류 문제 해결

Lambda 실행 오류가 문제인 경우에는 Lambda 함수에 대한 로깅 문을 생성해 CloudWatch 로그 파일에 CloudFront에서 함수의 실행을 모니터링하는 메시지를 작성한 다음 예상대로 작동하는지 확인하면 도움이 될 수 있습니다. 그런 다음 CloudWatch 로그 파일에서 해당 문을 검색해 함수가 작동 중인지 확인할 수 있습니다.

> **Note**
>
> Lambda@Edge 함수를 변경하지 않은 경우에도 Lambda 함수 실행 환경에 대한 업데이트가 영향을 줄 수 있으며, 실행 오류가 반환될 수 있습니다. 테스트 및 이후 버전으로의 마이그레이션에 대한 자세한 내용은 AWS Lambda 및 AWS Lambda@Edge 실행 환경에 대해 예정된 업데이트를 참조하십시오.

Lambda@Edge 리전 확인

Lambda@Edge 함수가 트래픽을 수신하는 리전을 보려면 AWS Management 콘솔에서 CloudFront 콘솔의 함수에 대한 지표를 확인하십시오. 지표는 각 AWS 리전별로 표시됩니다. 동일한 페이지에서 리전을 선택하고 해당 리전의 로그 파일을 확인하여 문제를 조사할 수 있습니다. CloudFront에서 Lambda 함수를 실행할 때 생성된 로그 파일을 확인하려면 올바른 AWS 리전의 CloudWatch 로그 파일을 검토해야 합니다.

CloudFront 콘솔의 모니터링 섹션에서 그래프를 보는 방법에 대한 자세한 내용은 Amazon CloudWatch를 사용하여 CloudFront 모니터링 (p. 416) 단원을 참조하십시오.

계정이 CloudWatch로 로그를 푸시하는지 확인

기본적으로 CloudFront는 잘못된 Lambda 함수 응답 로깅을 활성화하고 Lambda@Edge의 서비스 연결 역할 (p. 294) 중 하나를 수행해 해당 로그 파일을 CloudWatch로 푸시합니다. 잘못된 Lambda 함수 응답 로그

기능이 활성화되기 전에 CloudFront에 Lambda@Edge 함수를 추가한 경우에는 예를 들어, CloudFront 트리거 추가 등과 같이 다음에 Lambda@Edge 구성을 업데이트하는 경우 로깅이 활성화됩니다.

다음을 수행해 계정에 대해 CloudWatch로 로그 파일 푸시 기능이 활성화되어 있는지 확인할 수 있습니다.

- CloudWatch에 로그가 나타나는지 확인 Lambda@Edge 함수가 실행된 리전에서 로그 파일을 확인해야 합니다. 자세한 내용은 Lambda@Edge 리전 확인 (p. 313) 단원을 참조하십시오.
- IAM의 계정에 관련 서비스 연결 역할이 있는지 확인. 이렇게 하려면 https://console.aws.amazon.com/iam/에서 IAM 콘솔을 연 다음 역할을 선택해 계정에 대한 서비스 연결 역할 목록을 봅니다. `AWSServiceRoleForCloudFrontLogger` 역할을 찾습니다.

Lambda 함수용 CloudWatch 지표 및 CloudWatch Logs

CloudWatch 지표를 사용하여 Lambda 함수를 트리거하는 CloudFront 요청을 실시간으로 모니터링할 수 있습니다. 또한 CloudWatch Logs를 사용하여 집계 데이터를 얻을 수도 있습니다. 지표 또는 로그에 대한 추가 요금은 없습니다.

주제

- CloudWatch 지표 (p. 314)
- CloudWatch Logs (p. 314)

CloudWatch 지표

CloudFront 이벤트에 대한 트리거를 생성하면 Lambda가 자동으로 CloudWatch로 지표를 전송하기 시작합니다. 지표는 모든 Lambda 리전에서 사용할 수 있습니다. 각 지표의 이름은 `/aws/lambda/us-east-1.`*`function-name`*입니다. 여기서 *`function-name`*은 함수 생성 시 함수에 지정한 이름입니다. CloudWatch에서는 함수가 실행되는 위치와 가장 가까이 있는 리전으로 지표를 전송합니다.

CloudWatch 지표에 대한 자세한 내용은 Amazon CloudWatch 사용 설명서를 참조하십시오.

CloudWatch Logs

트리거를 생성하면 Lambda가 자동으로 Lambda 함수를 트리거하는 CloudFront 요청에 대한 데이터를 CloudWatch Logs로 전송하기 시작합니다. AWS Management 콘솔의 CloudFront 콘솔을 사용하여 로그 파일에 액세스하거나 CloudWatch Logs 도구를 사용하여 로그에 직접 액세스할 수 있습니다.

Lambda는 함수가 실행되는 위치와 가장 인접한 CloudWatch Logs 리전에서 CloudWatch Logs 로그 스트림을 생성합니다. 로그 그룹 이름의 형식은 `/aws/lambda/us-east-1.`*`function-name`*입니다. 여기서 *`function-name`*은 함수를 생성할 때 함수에 지정한 이름입니다.

> **Note**
>
> Lambda@Edge는 요청 볼륨과 로그 크기에 따라 로그를 제한합니다.

CloudFront에서 Lambda 함수를 실행할 때 생성된 로그 파일을 확인하려면 올바른 AWS 리전의 CloudWatch 로그 파일을 검토해야 합니다. Lambda@Edge 함수가 트래픽을 수신하는 리전을 보려면 AWS Management 콘솔에서 CloudFront 콘솔의 함수에 대한 지표 그래프를 확인하십시오. 지표는 각 AWS 리전별로 표시됩니다. 동일한 페이지에서 리전을 선택하고 해당 리전의 로그 파일을 확인하여 문제를 조사할 수 있습니다.

Lambda 함수와 함께 CloudWatch Logs를 사용하는 방법을 자세히 알아보려면 다음을 참조하십시오.

- CloudFront 콘솔의 모니터링 섹션에서 그래프를 보는 방법에 대한 자세한 내용은 Amazon CloudWatch를 사용하여 CloudFront 모니터링 (p. 416) 단원을 참조하십시오.

- CloudWatch Logs로 데이터를 전송하는 데 필요한 권한에 대한 자세한 내용은 IAM 사용 설명서의 Lambda@Edge에 대한 IAM 권한 및 역할 설정 단원을 참조하십시오.
- 로깅을 Lambda 함수에 추가하는 방법에 대한 자세한 내용은 AWS Lambda Developer Guide의 Node.js 함수 로깅 또는 Python 함수 로깅을 참조하십시오.
- CloudWatch Logs 할당량(이전에는 제한이라고 함)에 대한 자세한 내용은 Amazon CloudWatch Logs User Guide의 CloudWatch 로그 할당량을 참조하십시오.

Lambda@Edge 함수 및 복제본 삭제

CloudFront에서 함수의 복제본이 삭제된 경우에만 Lambda@Edge 함수를 삭제할 수 있습니다. 다음과 같은 상황에서 Lambda 함수 복제본이 자동으로 삭제됩니다.

- 함수의 복제본은 함수에 대한 마지막 연결을 모든 CloudFront 배포에서 제거한 후에 자동으로 삭제됩니다. 둘 이상의 배포에서 함수를 사용할 경우 마지막 배포에서 함수 연결이 해제된 후에만 복제본이 제거됩니다.
- 함수에 연결된 마지막 배포를 삭제한 후에도 복제본이 제거됩니다.

복제본은 일반적으로 몇 시간 내에 삭제됩니다.

버전에 연결된 CloudFront 배포가 없는 경우에는 특정 버전의 함수를 삭제할 수도 있습니다. 함수 버전에서 연결을 제거한 경우에는 보통 몇 시간 후에 함수를 삭제할 수 있습니다.

이때 복제본은 수동으로 삭제할 수 없습니다. 따라서 아직 사용 중인 복제본이 제거되어 오류가 발생하는 상황을 방지할 수 있습니다.

CloudFront 외부에서 함수 복제본을 사용하는 애플리케이션을 구축하지 마십시오. 이러한 복제본은 배포와의 연결이 제거되거나 배포 자체가 삭제될 때마다 삭제됩니다. 이 경우 외부 애플리케이션에서 사용하는 복제본이 경고도 없이 제거되어 오류가 발생할 수 있습니다.

Lambda@Edge 이벤트 구조

다음 주제에서는 트리거될 때 CloudFront가 Lambda@Edge 함수에 전달하는 요청 및 응답 이벤트 객체에 대해 설명합니다.

주제

- 동적 오리진 선택 (p. 315)
- 이벤트 요청 (p. 316)
- 응답 이벤트 (p. 321)

동적 오리진 선택

캐시 동작의 경로 패턴 (p. 44)을 사용하여 `images/*.jpg`와 같이 요청된 객체의 경로와 이름을 기반으로 오리진에 요청을 라우팅할 수 있습니다. 또한 Lambda@Edge를 사용하여 요청 헤더의 값과 같은 기타 특성을 기반으로 오리진에 요청을 라우팅할 수 있습니다.

여러 경우에 이러한 동적 오리진 선택이 유용할 수 있습니다. 예를 들어, 각기 다른 지리 영역에 있는 오리진에 걸쳐 요청을 배포하여 전역 로드 밸런싱에 도움이 될 수 있습니다. 또는 봇 처리, SEO 최적화, 인증 등과 같은 특정 기능을 서비스하는 각기 다른 오리진에 요청을 선택적으로 라우팅할 수 있습니다. 이 기능을 사용하는 방법을 보여주는 코드 예제는 콘텐츠 기반 동적 오리진 선택 - 예제 (p. 345) 단원을 참조하십시오.

CloudFront 오리진 요청 이벤트에서 이벤트 구조의 오리진 `origin` 객체에는 요청이 경로 패턴을 기반으로 라우팅될 오리진에 대한 정보가 들어 있습니다. `origin` 객체의 값을 업데이트하여 요청을 다른 오리진으로 라우팅할 수 있습니다. `origin` 객체를 업데이트할 때 배포에서 오리진을 정의할 필요가 없습니다. 또한 Amazon S3 오리진 객체를 사용자 지정 오리진 객체로 바꿀 수 있으며, 그 반대의 경우도 마찬가지입니다. 요청당 하나의 오리진만 지정할 수 있습니다. 즉, 사용자 지정 오리진 또는 Amazon S3 오리진 중 하나를 지정할 수 있지만 둘 모두 지정할 수는 없습니다.

이벤트 요청

다음 주제에서는 CloudFront가 최종 사용자 및 오리진 요청 이벤트 (p. 302)에 대한 Lambda 함수에 전달하는 객체의 구조를 보여줍니다. 이 예제에서는 본문이 없는 `GET` 요청을 보여줍니다. 다음 예제는 최종 사용자 및 오리진 요청 이벤트에서 가능한 모든 필드의 목록입니다.

주제

- 최종 사용자 요청 예제 (p. 316)
- 오리진 요청 예제 (p. 317)
- 요청 이벤트 필드 (p. 318)

최종 사용자 요청 예제

다음 예제에서는 최종 사용자 요청 이벤트 객체를 보여줍니다.

```
{
  "Records": [
    {
      "cf": {
        "config": {
          "distributionDomainName": "d111111abcdef8.cloudfront.net",
          "distributionId": "EDFDVBD6EXAMPLE",
          "eventType": "viewer-request",
          "requestId": "4TyzHTaYWb1GX1qTfsHhEqV6HUDd_BzoBZnwfnvQc_1oF26ClkoUSEQ=="
        },
        "request": {
          "clientIp": "203.0.113.178",
          "headers": {
            "host": [
              {
                "key": "Host",
                "value": "d111111abcdef8.cloudfront.net"
              }
            ],
            "user-agent": [
              {
                "key": "User-Agent",
                "value": "curl/7.66.0"
              }
            ],
            "accept": [
              {
                "key": "accept",
                "value": "*/*"
              }
            ]
          },
          "method": "GET",
          "querystring": "",
          "uri": "/"
        }
      }
    }
```

```
  ]
}
```

오리진 요청 예제

다음 예제에서는 오리진 요청 이벤트 객체를 보여줍니다.

```
{
  "Records": [
    {
      "cf": {
        "config": {
          "distributionDomainName": "d111111abcdef8.cloudfront.net",
          "distributionId": "EDFDVBD6EXAMPLE",
          "eventType": "origin-request",
          "requestId": "4TyzHTaYWb1GX1qTfsHhEqV6HUDd_BzoBZnwfnvQc_1oF26ClkoUSEQ=="
        },
        "request": {
          "clientIp": "203.0.113.178",
          "headers": {
            "x-forwarded-for": [
              {
                "key": "X-Forwarded-For",
                "value": "203.0.113.178"
              }
            ],
            "user-agent": [
              {
                "key": "User-Agent",
                "value": "Amazon CloudFront"
              }
            ],
            "via": [
              {
                "key": "Via",
                "value": "2.0 2afae0d44e2540f472c0635ab62c232b.cloudfront.net (CloudFront)"
              }
            ],
            "host": [
              {
                "key": "Host",
                "value": "example.org"
              }
            ],
            "cache-control": [
              {
                "key": "Cache-Control",
                "value": "no-cache, cf-no-cache"
              }
            ]
          },
          "method": "GET",
          "origin": {
            "custom": {
              "customHeaders": {},
              "domainName": "example.org",
              "keepaliveTimeout": 5,
              "path": "",
              "port": 443,
              "protocol": "https",
              "readTimeout": 30,
              "sslProtocols": [
                "TLSv1",
                "TLSv1.1",
```

```
                "TLSv1.2"
              ]
            }
          },
          "querystring": "",
          "uri": "/"
        }
      }
    }
  ]
}
```

요청 이벤트 필드

요청 이벤트 객체 데이터는 두 개의 하위 객체(`config`(`Records.cf.config`) 및 `request`(`Records.cf.request`))에 포함되어 있습니다. 다음 목록에서는 각 하위 객체의 필드에 대해 설명합니다.

구성 객체의 필드

다음 표에서는 `config` 객체(`Records.cf.config`)의 필드에 대해 설명합니다.

`distributionDomainName`(읽기 전용)

요청과 연결된 배포의 도메인 이름입니다.

`distributionID`(읽기 전용)

요청과 관련이 있는 배포의 ID입니다.

`eventType`(읽기 전용)

요청(`viewer-request` 또는 `origin-request`)과 관련된 트리거의 유형입니다.

`requestId`(읽기 전용)

최종 사용자 요청/CloudFront 요청을 고유하게 식별하는 암호화된 문자열입니다. 또한 `requestId` 값은 CloudFront 액세스 로그에 `x-edge-request-id`로 표시됩니다. 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 및 웹 배포 로그 파일 형식 (p. 399) 단원을 참조하십시오.

요청 객체의 필드

다음 표에서는 `request` 객체(`Records.cf.request`)의 필드에 대해 설명합니다.

`clientIp`(읽기 전용)

요청을 생성한 최종 사용자의 IP 주소입니다. 최종 사용자가 HTTP 프록시 또는 로드 밸런서를 사용하여 요청을 전송한 경우 이 값은 프록시 또는 로드 밸런서의 IP 주소입니다.

headers(읽기/쓰기)

요청의 헤더입니다. 다음을 참조하십시오.

- `headers` 객체 내 키는 표준 HTTP 헤더 이름의 소문자 버전입니다. 소문자 키를 사용하여 대/소문자를 구분하지 않고 헤더 값에 액세스할 수 있습니다.
- 각 헤더 객체(예: `headers["accept"]` 또는 `headers["host"]`)는 키-값 페어의 어레이입니다. 해당 헤더에 대해 어레이에는 요청의 각 값에 대한 하나의 키-값 페어를 포함합니다.
- `key`에는 HTTP 요청에 표시되는 헤더의 대/소문자 구분 이름을 포함합니다(예: `Host`, `User-Agent`, `X-Forwarded-For` 등).
- `value`에는 HTTP 요청에 표시되는 헤더 값을 포함합니다.
- Lambda 함수가 요청 헤더를 추가하거나 수정하고 사용자가 헤더 `key` 필드를 포함시키지 않으면, Lambda@Edge는 사용자가 입력한 헤더 이름을 사용하여 헤더 `key`를 자동으로 삽입합니다. 헤더 이

름의 형식을 어떻게 지정했든 상관없이, 자동으로 삽입된 헤더 키는 각 부분의 첫 문자가 대문자로 지정되고 각 부분이 하이픈(-)으로 구분됩니다.

예를 들어 헤더를 다음과 같이 헤더 `key` 없이 추가할 수 있습니다.

```
"user-agent": [
  {
    "value": "ExampleCustomUserAgent/1.X.0"
  }
]
```

이 예제에서는 Lambda@Edge가 자동으로 `"key": "User-Agent"`를 삽입합니다.

헤더 사용 시 제한 사항에 대한 자세한 내용은 헤더 (p. 358) 단원을 참조하십시오.

`method`(읽기 전용)

요청의 HTTP 메서드.

`querystring`(읽기/쓰기)

요청의 쿼리 문자열입니다(있는 경우). 요청에 쿼리 문자열이 포함되지 않더라도 이벤트 객체에는 여전히 빈 값과 함께 `querystring`이 포함됩니다. 쿼리 문자열에 대한 자세한 내용은 쿼리 문자열 파라미터 기반의 콘텐츠 캐싱 (p. 196) 단원을 참조하십시오.

`uri`(읽기/쓰기)

요청된 객체의 상대 경로입니다. Lambda 함수가 `uri` 값을 수정하는 경우 다음 사항에 유의하십시오.

- 새 `uri` 값은 슬래시(/) 로 시작해야 합니다.
- 함수가 `uri` 값을 변경하는 경우 이로 인해 최종 사용자가 요청 중인 객체가 변경됩니다.
- 함수가 `uri` 값을 변경하는 경우 이로 인해 요청 또는 요청이 전송되는 오리진에 대한 캐시 동작이 변경되지 않습니다.

`body`(읽기/쓰기)

HTTP 요청의 본문입니다. `body` 구조에는 다음과 같은 필드가 포함될 수 있습니다.

`inputTruncated`(읽기 전용)

Lambda@Edge가 본문을 잘랐는지 여부를 나타내는 부울 플래그. 자세한 내용은 본문 포함 옵션이 적용된 요청 본문에 대한 크기 할당량 (p. 362) 단원을 참조하십시오.

`action`(읽기/쓰기)

본문을 사용해 수행하고자 한 작업. `action`의 옵션은 다음과 같습니다.

- `read-only`: 이 값이 기본값입니다. Lambda 함수에서 응답이 반환되었는데 `action`이 read-only이면, Lambda@Edge는 `encoding` 또는 `data`에 대한 모든 변경 사항을 무시합니다.
- `replace`: 오리진으로 전송된 본문을 바꾸려는 경우 이 옵션을 지정합니다.

`encoding`(읽기/쓰기)

본문에 대한 인코딩. Lambda@Edge가 Lambda 함수에 본문을 노출시키는 경우 먼저 본문을 base64-encoding으로 변환합니다. 본문을 바꾸기 위해 `action`에 `replace`를 선택한 경우 `base64`(기본값) 또는 `text` 인코딩을 사용하도록 선택할 수 있습니다. `encoding`을 `base64`로 지정하지만 본문이 유효한 base64가 아닌 경우 CloudFront에서는 오류를 반환합니다.

`data`(읽기/쓰기)

요청 본문의 내용

`origin`(읽기/쓰기) (오리진 이벤트만 해당)

요청을 전송할 오리진입니다. `origin` 구조에는 하나의 오리진만 포함되어야 합니다. 즉, 오리진은 사용자 지정 오리진 또는 Amazon S3 오리진일 수 있습니다. 오리진 구조에는 다음과 같은 필드가 포함될 수 있습니다.

customHeaders(읽기/쓰기) (사용자 지정 오리진 및 Amazon S3 오리진)

각 사용자 지정 헤더에 대한 헤더 이름 및 값 페어를 지정하여 요청을 포함한 사용자 지정 헤더를 포함할 수 있습니다. 블랙리스트에 등록된 헤더는 추가할 수 없으며 동일한 이름의 헤더가 `Records.cf.request.headers`에 존재할 수 없습니다. 요청 헤더에 대한 노트 (p. 318)는 사용자 지정 헤더에도 적용됩니다. 자세한 내용은 CloudFront가 오리진 요청에 추가할 수 없는 사용자 지정 헤더 (p. 249) 및 블랙리스트 헤더 (p. 358) 단원을 참조하십시오.

domainName(읽기/쓰기) (사용자 지정 오리진 및 Amazon S3 오리진)

오리진의 도메인 이름입니다. 도메인 이름은 비워둘 수 없습니다.

- 사용자 지정 오리진의 경우 – DNS 도메인 이름을 지정합니다(예: `www.example.com`). 도메인 이름에는 콜론(:)이 포함될 수 없으며 IP 주소가 될 수 없습니다. 도메인 이름은 최대 253자일 수 있습니다.
- Amazon S3 오리진의 경우 – Amazon S3 버킷의 DNS 도메인 이름을 지정합니다(예: `awsexamplebucket.s3.eu-west-1.amazonaws.com`). 이름은 최대 128자여야 하며, 모두 소문자여야 합니다.

path(읽기/쓰기) (사용자 지정 오리진 및 Amazon S3 오리진)

요청이 콘텐츠를 찾아야 하는 오리진의 디렉터리 경로입니다. 경로는 슬래시(/)로 시작해야 하지만 슬래시 하나로 끝나지 않아야 합니다(예: `example-path/`와 같이 끝나면 안 됨). 사용자 지정 오리진의 경우만, 경로는 URL로 인코딩되어야 하고 최대 길이는 255자여야 합니다.

keepaliveTimeout(읽기/쓰기) (사용자 지정 오리진만 해당)

CloudFront가 응답의 마지막 패킷을 수신한 후 오리진 연결을 유지하기 위해 시도해야 하는 시간(초)입니다. 값은 1~60(경계값 포함) 사이의 숫자여야 합니다.

port(읽기/쓰기) (사용자 지정 오리진만 해당)

CloudFront가 사용자 지정 오리진에서 연결해야 하는 포트입니다. 포트는 80, 443 또는 1024~65535(경계값 포함) 사이의 숫자여야 합니다.

protocol(읽기/쓰기) (사용자 지정 오리진만 해당)

오리진에 연결할 때 CloudFront가 사용해야 하는 연결 프로토콜입니다. `http` 또는 `https` 값을 가질 수 있습니다.

readTimeout(읽기/쓰기) (사용자 지정 오리진만 해당)

요청을 오리진으로 전송한 후 CloudFront가 응답을 대기해야 하는 시간(초)입니다. 또한 다음 패킷을 수신하기 전에 응답 패킷을 수신한 후 CloudFront가 대기해야 하는 시간도 지정합니다. 값은 4~60(경계값 포함) 사이의 숫자여야 합니다.

sslProtocols(읽기/쓰기) (사용자 지정 오리진만 해당)

오리진과의 HTTPS 연결을 설정할 때 CloudFront가 사용할 수 있는 최소 SSL/TLS 프로토콜입니다. 값은 `TLSv1.2`, `TLSv1.1`, `TLSv1` 또는 `SSLv3` 중 하나일 수 있습니다.

authMethod(읽기/쓰기) (Amazon S3 오리진만 해당)

OAI(원본 액세스 ID) (p. 176)를 사용하는 경우 이 필드를 `origin-access-identity`로 설정하거나 OAI를 사용하지 않는 경우 `none`으로 설정합니다. `authMethod`를 `origin-access-identity`로 설정하면 다음과 같은 몇 가지 요구 사항이 있습니다.

- `region`을 지정해야 합니다(다음 필드 참조).
- 하나의 Amazon S3 오리진에서 다른 오리진으로 요청을 변경할 때 동일한 OAI를 사용해야 합니다.
- 사용자 지정 오리진에서 Amazon S3 오리진으로 요청을 변경할 때 OAI를 사용할 수 없습니다.

region(읽기/쓰기) (Amazon S3 오리진만 해당)

Amazon S3 버킷의 AWS 리전입니다. 이 작업은 `authMethod`를 `origin-access-identity`로 설정한 경우에만 필요합니다.

응답 이벤트

다음 주제에서는 CloudFront가 최종 사용자 및 오리진 응답 이벤트 (p. 302)에 대한 Lambda 함수에 전달하는 객체의 구조를 보여줍니다. 다음 예제는 최종 사용자 및 오리진 응답 이벤트에서 가능한 모든 필드의 목록입니다.

주제

- 오리진 응답 예제 (p. 321)
- 최종 사용자 응답 예제 (p. 323)
- 응답 이벤트 필드 (p. 325)

오리진 응답 예제

다음 예제에서는 오리진 응답 이벤트 객체를 보여줍니다.

```
{
  "Records": [
    {
      "cf": {
        "config": {
          "distributionDomainName": "d111111abcdef8.cloudfront.net",
          "distributionId": "EDFDVBD6EXAMPLE",
          "eventType": "origin-response",
          "requestId": "4TyzHTaYWb1GX1qTfsHhEqV6HUDd_BzoBZnwfnvQc_1oF26ClkoUSEQ=="
        },
        "request": {
          "clientIp": "203.0.113.178",
          "headers": {
            "x-forwarded-for": [
              {
                "key": "X-Forwarded-For",
                "value": "203.0.113.178"
              }
            ],
            "user-agent": [
              {
                "key": "User-Agent",
                "value": "Amazon CloudFront"
              }
            ],
            "via": [
              {
                "key": "Via",
                "value": "2.0 8f22423015641505b8c857a37450d6c0.cloudfront.net (CloudFront)"
              }
            ],
            "host": [
              {
                "key": "Host",
                "value": "example.org"
              }
            ],
            "cache-control": [
              {
                "key": "Cache-Control",
                "value": "no-cache, cf-no-cache"
              }
            ]
          },
          "method": "GET",
```

```
          "origin": {
            "custom": {
              "customHeaders": {},
              "domainName": "example.org",
              "keepaliveTimeout": 5,
              "path": "",
              "port": 443,
              "protocol": "https",
              "readTimeout": 30,
              "sslProtocols": [
                "TLSv1",
                "TLSv1.1",
                "TLSv1.2"
              ]
            }
          },
          "querystring": "",
          "uri": "/"
        },
        "response": {
          "headers": {
            "access-control-allow-credentials": [
              {
                "key": "Access-Control-Allow-Credentials",
                "value": "true"
              }
            ],
            "access-control-allow-origin": [
              {
                "key": "Access-Control-Allow-Origin",
                "value": "*"
              }
            ],
            "date": [
              {
                "key": "Date",
                "value": "Mon, 13 Jan 2020 20:12:38 GMT"
              }
            ],
            "referrer-policy": [
              {
                "key": "Referrer-Policy",
                "value": "no-referrer-when-downgrade"
              }
            ],
            "server": [
              {
                "key": "Server",
                "value": "ExampleCustomOriginServer"
              }
            ],
            "x-content-type-options": [
              {
                "key": "X-Content-Type-Options",
                "value": "nosniff"
              }
            ],
            "x-frame-options": [
              {
                "key": "X-Frame-Options",
                "value": "DENY"
              }
            ],
            "x-xss-protection": [
              {
                "key": "X-XSS-Protection",
```

```
                "value": "1; mode=block"
              }
            ],
            "content-type": [
              {
                "key": "Content-Type",
                "value": "text/html; charset=utf-8"
              }
            ],
            "content-length": [
              {
                "key": "Content-Length",
                "value": "9593"
              }
            ]
          },
          "status": "200",
          "statusDescription": "OK"
        }
      }
    }
  ]
}
```

최종 사용자 응답 예제

다음 예제에서는 최종 사용자 응답 이벤트 객체를 보여줍니다.

```
{
  "Records": [
    {
      "cf": {
        "config": {
          "distributionDomainName": "d111111abcdef8.cloudfront.net",
          "distributionId": "EDFDVBD6EXAMPLE",
          "eventType": "viewer-response",
          "requestId": "4TyzHTaYWb1GX1qTfsHhEqV6HUDd_BzoBZnwfnvQc_1oF26ClkoUSEQ==="
        },
        "request": {
          "clientIp": "203.0.113.178",
          "headers": {
            "host": [
              {
                "key": "Host",
                "value": "d111111abcdef8.cloudfront.net"
              }
            ],
            "user-agent": [
              {
                "key": "User-Agent",
                "value": "curl/7.66.0"
              }
            ],
            "accept": [
              {
                "key": "accept",
                "value": "*/*"
              }
            ]
          },
          "method": "GET",
          "querystring": "",
          "uri": "/"
        },
```

```
        "response": {
          "headers": {
            "access-control-allow-credentials": [
              {
                "key": "Access-Control-Allow-Credentials",
                "value": "true"
              }
            ],
            "access-control-allow-origin": [
              {
                "key": "Access-Control-Allow-Origin",
                "value": "*"
              }
            ],
            "date": [
              {
                "key": "Date",
                "value": "Mon, 13 Jan 2020 20:14:56 GMT"
              }
            ],
            "referrer-policy": [
              {
                "key": "Referrer-Policy",
                "value": "no-referrer-when-downgrade"
              }
            ],
            "server": [
              {
                "key": "Server",
                "value": "ExampleCustomOriginServer"
              }
            ],
            "x-content-type-options": [
              {
                "key": "X-Content-Type-Options",
                "value": "nosniff"
              }
            ],
            "x-frame-options": [
              {
                "key": "X-Frame-Options",
                "value": "DENY"
              }
            ],
            "x-xss-protection": [
              {
                "key": "X-XSS-Protection",
                "value": "1; mode=block"
              }
            ],
            "age": [
              {
                "key": "Age",
                "value": "2402"
              }
            ],
            "content-type": [
              {
                "key": "Content-Type",
                "value": "text/html; charset=utf-8"
              }
            ],
            "content-length": [
              {
                "key": "Content-Length",
                "value": "9593"
```

```
                }
              ]
            },
            "status": "200",
            "statusDescription": "OK"
          }
        }
      }
    ]
}
```

응답 이벤트 필드

응답 이벤트 객체 데이터는 세 개의 하위 객체(`config`(`Records.cf.config`), `request`(`Records.cf.request`) 및 `response`(`Records.cf.response`))에 포함되어 있습니다. 요청 객체의 필드에 대한 자세한 내용은 요청 객체의 필드 (p. 318) 단원을 참조하십시오. 다음 목록에서는 `config` 및 `response` 하위 객체의 필드에 대해 설명합니다.

구성 객체의 필드

다음 표에서는 `config` 객체(`Records.cf.config`)의 필드에 대해 설명합니다.

`distributionDomainName`(읽기 전용)

응답과 연결된 배포의 도메인 이름입니다.

`distributionID`(읽기 전용)

응답과 관련이 있는 배포의 ID입니다.

`eventType`(읽기 전용)

응답과 관련된 트리거의 유형입니다(`origin-response` 또는 `viewer-response`).

`requestId`(읽기 전용)

이 응답이 연결된 최종 사용자 요청/CloudFront 요청을 고유하게 식별하는 암호화된 문자열입니다. 또한 `requestId` 값은 CloudFront 액세스 로그에 `x-edge-request-id`로 표시됩니다. 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 및 웹 배포 로그 파일 형식 (p. 399) 단원을 참조하십시오.

응답 객체의 필드

다음 표에서는 `response` 객체(`Records.cf.response`)의 필드에 대해 설명합니다. Lambda@Edge 함수를 사용하여 HTTP 응답을 생성하는 방법에 대한 자세한 내용은 요청 트리거에서 HTTP 응답 생성 (p. 326) 단원을 참조하십시오.

`headers`(읽기/쓰기)

응답의 헤더입니다. 다음을 참조하십시오.

- `headers` 객체 내 키는 표준 HTTP 헤더 이름의 소문자 버전입니다. 소문자 키를 사용하여 대/소문자를 구분하지 않고 헤더 값에 액세스할 수 있습니다.
- 각 헤더 객체(예: `headers["content-type"]` 또는 `headers["content-length"]`)는 키-값 페어의 어레이입니다. 해당 헤더에 대해 어레이에는 응답의 각 값에 대한 하나의 키-값 페어를 포함합니다.
- `key`에는 HTTP 응답에 표시되는 헤더의 대/소문자 구분 이름을 포함합니다(예: `Content-Type`, `Content-Length` 등).
- `value`에는 HTTP 응답에 표시되는 헤더 값을 포함합니다.
- Lambda 함수가 응답 헤더를 추가하거나 수정하고 사용자가 헤더 `key` 필드를 포함시키지 않으면, Lambda@Edge는 사용자가 입력한 헤더 이름을 사용하여 헤더 `key`를 자동으로 삽입합니다. 헤더 이

름의 형식을 어떻게 지정했든 상관없이, 자동으로 삽입된 헤더 키는 각 부분의 첫 문자가 대문자로 지정되고 각 부분이 하이픈(-)으로 구분됩니다.

예를 들어 헤더를 다음과 같이 헤더 `key` 없이 추가할 수 있습니다.

```
"content-type": [
  {
    "value": "text/html;charset=UTF-8"
  }
]
```

이 예제에서는 Lambda@Edge가 자동으로 `"key": "Content-Type"`을 삽입합니다.

헤더 사용 시 제한 사항에 대한 자세한 내용은 헤더 (p. 358) 단원을 참조하십시오.

`status`

응답의 HTTP 상태 코드입니다.

`statusDescription`

응답의 HTTP 상태 설명입니다.

요청 및 응답으로 작업

Lambda@Edge를 사용하여 다음과 같은 여러 가지 방법으로 요청과 응답으로 작업할 수 있습니다.

- 요청 트리거에서 HTTP 응답 생성 (p. 326)
- 오리진 응답 트리거에서 HTTP 응답 업데이트 (p. 329)
- 본문 포함 옵션을 선택해 요청 본문에 액세스 (p. 329)
- Lambda@Edge 함수와 오리진 장애 조치 사용 (p. 326)

Lambda@Edge 함수와 오리진 장애 조치 사용

예를 들어 고가용성 확보를 위해 구성하는 오리진 장애 조치에 대하여 오리진 그룹과 함께 설정한 CloudFront 배포와 함께 Lambda@Edge 함수를 사용할 수 있습니다. 오리진 그룹과 함께 Lambda 함수를 사용하려면, 캐시 동작을 생성할 때 오리진 그룹에 대한 오리진 요청 또는 오리진 응답 트리거에 함수를 지정합니다.

자세한 정보는 다음 단원을 참조하십시오.

- 오리진 그룹 만들기: 오리진 그룹 만들기 (p. 212)
- 오리진 장애 조치가 Lambda@Edge와 함께 작동하는 방식: Lambda@Edge 함수와 함께 오리진 장애 조치 사용 (p. 212)

요청 트리거에서 HTTP 응답 생성

CloudFront가 요청을 수신할 때 Lambda 함수를 사용하여 CloudFront가 오리진에 응답을 전달하지 않고 최종 사용자에게 직접 반환하는 HTTP 응답을 생성할 수 있습니다. HTTP 응답을 생성하면 오리진에 대한 로드가 감소하고 일반적으로 최종 사용자에 대한 지연 시간 또한 감소합니다.

HTTP 응답 생성의 몇 가지 일반 시나리오는 다음을 포함합니다.

- 최종 사용자에게 소규모 웹 페이지 반환
- HTTP 301 또는 302 상태 코드를 반환하여 다른 웹 페이지로 사용자를 리디렉션

- 사용자가 인증되지 않았을 때 최종 사용자에게 HTTP 401 상태 코드 반환

Lambda@Edge 함수는 다음 CloudFront 이벤트가 발생할 때 HTTP 응답을 생성할 수 있습니다.

최종 사용자 요청 이벤트

최종 사용자 요청 이벤트에 의해 함수가 트리거될 때 CloudFront는 최종 사용자에게 응답을 반환하고 이를 캐시하지 않습니다.

오리진 요청 이벤트

오리진 요청 이벤트에 의해 함수가 트리거될 때 CloudFront는 함수에 의해 이전에 생성된 응답에 대한 엣지 캐시를 확인합니다.

- 응답이 캐시에 있는 경우 함수는 실행되지 않고 CloudFront는 캐싱된 응답을 최종 사용자에게 반환합니다.
- 응답이 캐시에 없는 경우 함수는 실행되고 CloudFront는 캐싱된 응답을 최종 사용자에게 반환하며 이를 캐시합니다.

HTTP 응답을 생성하는 몇 가지 샘플 코드를 보려면 Lambda@Edge 함수 예제 (p. 330) 단원을 참조하십시오. 응답 트리거에서 HTTP 응답을 바꿀 수도 있습니다. 자세한 내용은 오리진 응답 트리거에서 HTTP 응답 업데이트 (p. 329) 단원을 참조하십시오.

프로그래밍 모델

이 단원에서는 Lambda@Edge를 사용하여 HTTP 응답을 생성하기 위한 프로그래밍 모델을 설명합니다.

주제

- 응답 객체 (p. 327)
- 오류 (p. 328)
- 필수 필드 (p. 329)

응답 객체

`result` 메서드의 `callback`파라미터로 반환한 응답은 다음 구조를 가져야 합니다(`status` 필드만이 필요함).

```
const response = {
    body: 'content',
    bodyEncoding: 'text' | 'base64',
    headers: {
        'header name in lowercase': [{
            key: 'header name in standard case',
            value: 'header value'
         }],
         ...
    },
    status: 'HTTP status code',
    statusDescription: 'status description'
};
```

응답 객체에는 다음 값이 포함될 수 있습니다.

`body`

CloudFront가 생성된 응답에서 반환하고자 하는 본문입니다.

bodyEncoding

`body`에서 지정한 값에 대한 인코딩입니다. 유일하게 유효한 인코딩은 `text`과 `base64`입니다. `response` 객체에 `body`를 포함했지만 `bodyEncoding`을 누락한 경우 CloudFront는 본문을 텍스트로 처리합니다.

`bodyEncoding`을 `base64`로 지정했지만 본문이 유효한 base64가 아닌 경우 CloudFront는 오류를 반환합니다.

headers

CloudFront가 생성된 응답에서 반환하고자 하는 헤더입니다. 다음을 참조하십시오.

- `headers` 객체 내 키는 표준 HTTP 헤더 이름의 소문자 버전입니다. 소문자 키를 사용하여 대/소문자를 구분하지 않고 헤더 값에 액세스할 수 있습니다.
- 각 헤더(예: `headers["accept"]` 또는 `headers["host"]`)는 키-값 페어의 어레이입니다. 해당 헤더에 대해 어레이에는 생성된 응답의 각 값에 대한 하나의 키-값 페어를 포함합니다.
- `key`(선택사항)는 HTTP 요청에 표시되는 헤더의 대/소문자 구분 이름입니다(예: `accept` 또는 `host`).
- `value`를 헤더 값으로 지정합니다.
- 키-값 페어의 헤더 키 부분을 포함시키지 않으면 Lambda@Edge는 사용자가 입력한 헤더 이름을 사용하여 헤더 키를 자동으로 삽입합니다. 헤더 이름의 형식을 어떻게 지정했든 상관없이, 자동으로 삽입된 헤더 키는 각 부분의 첫 문자가 대문자로 지정되고 각 부분이 하이픈(-)으로 구분됩니다.

  예를 들어 헤더를 다음과 같이 헤더 키 없이 추가할 수 있습니다. `'content-type': [{ value: 'text/html;charset=UTF-8' }]`

  이 예제에서 Lambda@Edge는 다음과 같은 헤더 키를 생성합니다. `Content-Type`

헤더 사용 시 제한 사항에 대한 자세한 내용은 헤더 (p. 358) 단원을 참조하십시오.

status

CloudFront가 다음에 대해 사용하고자 하는 HTTP 상태 코드입니다.

- 응답 시 반환
- 오리진 요청 이벤트에 의해 트리거된 함수에 의해 응답이 생성되었을 때의 CloudFront 엣지 캐시에 있는 캐시
- CloudFront 액세스 로그 구성 및 사용 (p. 393)에 로깅

`status` 값이 200~599가 아닌 경우 CloudFront는 최종 사용자에게 오류를 반환합니다.

statusDescription

CloudFront가 응답에서 반환하고 HTTP 상태 코드를 포함하고자 하는 설명입니다. HTTP 상태 코드 200에 대해 `OK`와 같은 표준 설명을 사용할 필요는 없습니다.

오류

다음은 생성된 HTTP 응답에서 발생할 수 있는 오류입니다.

본문을 포함하고 상태에 대해 204(콘텐츠 없음)을 지정한 응답

최종 사용자 요청 이벤트에 의해 함수가 트리거될 때 CloudFront는 다음 두 가지가 모두 true일 경우 HTTP 502 상태 코드(잘못된 게이트웨이)를 반환합니다.

- `status`의 기본값이 204(콘텐츠 없음)임
- 응답에 `body`에 대한 값이 포함

이는 Lambda@Edge가 `HTTP 204` 응답에 메시지 본문이 포함될 필요가 없다고 명시한 RFC 2616의 선택적인 제한을 부과하기 때문입니다.

생성된 응답의 크기 제한

Lambda 함수가 생성하는 응답의 최대 크기는 함수를 트리거한 이벤트에 따라 다릅니다.

- 최종 사용자 요청 이벤트 – 40KB
- 오리진 요청 이벤트 – 1MB

응답이 허용된 크기보다 큰 경우 CloudFront는 최종 사용자에게 HTTP 502 상태 코드(잘못된 게이트웨이)를 반환합니다.

필수 필드

`status` 필드는 필수입니다.

다른 모든 필드는 선택 사항입니다.

오리진 응답 트리거에서 HTTP 응답 업데이트

CloudFront가 오리진 서버에서 HTTP 응답을 수신할 때 오리진 응답 트리거가 캐시 동작과 연결된 경우 HTTP 응답을 수정하여 오리진에서 반환된 것을 재정의할 수 있습니다.

HTTP 응답 업데이트의 몇 가지 일반 시나리오는 다음을 포함합니다.

- 오리진이 오류 상태 코드(4xx 또는 5xx)를 반환할 때 상태를 변경하여 HTTP 200 상태 코드를 설정하고 정적 본문 콘텐츠를 생성하여 최종 사용자에게 반환합니다. 샘플 코드에 대한 내용은 예제: 오리진 응답 트리거를 사용하여 오류 상태 코드를 200-OK로 업데이트 (p. 352)를 참조하십시오.
- 오리진이 오류 상태 코드(4xx 또는 5xx)를 반환할 때 상태를 변경하여 HTTP 301 또는 HTTP 302 상태 코드를 설정하고 사용자를 다른 웹 사이트로 리디렉션합니다. 샘플 코드에 대한 내용은 예제: 오리진 응답 트리거를 사용하여 오류 상태 코드를 302-Found로 업데이트 (p. 353)를 참조하십시오.

최종 사용자 및 오리진 요청 이벤트에서 HTTP 응답을 바꿀 수도 있습니다. 자세한 내용은 요청 트리거에서 HTTP 응답 생성 (p. 326) 단원을 참조하십시오.

HTTP 응답을 사용하여 작업 중일 때 Lambda@Edge는 오리진 서버에 의해 오리진 응답 트리거로 반환되는 HTML 본문을 공개하지 않습니다. 원하는 값으로 설정하여 정적 콘텐츠 본문을 생성하거나 값을 비어 있음으로 설정하여 함수 내부의 본문을 제거할 수 있습니다. 함수의 본문 필드를 업데이트하지 않은 경우 오리진 서버에서 반환된 원래 본문이 최종 사용자에게 반환됩니다.

본문 포함 옵션을 선택해 요청 본문에 액세스

이제 Lambda 함수에서 액세스할 수 있도록 Lambda@Edge가 쓰기 가능한 HTTP 메서드(POST, PUT, DELETE 등)에 대한 요청에 본문을 노출하도록 선택할 수 있습니다. 읽기 전용 액세스 권한을 선택하거나 본문을 바꾸도록 지정할 수 있습니다.

이 옵션을 활성화하려면 함수에 대해 최종 사용자 요청 또는 오리진 요청 이벤트에 대해 실행되는 CloudFront 트리거를 생성할 때 본문 포함을 선택합니다. 자세한 내용은 Lambda@Edge 함수에 대한 트리거 추가 (p. 302) 단원을 참조하고, 함수에서 본문 포함을 사용하는 방법을 알아보려면 Lambda@Edge 이벤트 구조 (p. 315) 단원을 참조하십시오.

이 기능을 사용하려는 시나리오는 다음과 같을 수 있습니다.

- 오리진 서버로 고객 입력 데이터를 다시 보내지 않는 웹 양식(예: "문의처") 처리
- 최종 사용자의 브라우저가 보낸 웹 비콘 데이터를 수집해 엣지에서 처리

샘플 코드에 대한 내용은 Lambda@Edge 함수 예제 (p. 330)를 참조하십시오.

> **Note**
>
> 요청 본문이 크면 Lambda@Edge가 자릅니다. 최대 크기 및 잘림에 대한 자세한 내용은 본문 포함 옵션이 적용된 본문에 대한 최대 크기 (p. 363) 단원을 참조하십시오.

Lambda@Edge 함수 예제

CloudFront에서 Lambda 함수를 사용하는 예제는 다음 단원을 참조하십시오.

> **Note**
>
> Node.js 함수의 경우 각 함수는 `callback` 파라미터를 호출하여 요청을 성공적으로 처리하거나 응답을 반환해야 합니다. 자세한 내용은 Lambda@Edge 함수 작성 및 생성 (p. 298) 단원을 참조하십시오.

주제

- 일반 예제 (p. 330)
- 응답 생성 - 예제 (p. 333)
- 쿼리 문자열 처리 - 예제 (p. 337)
- 국가 또는 디바이스 유형 헤더에 따른 콘텐츠 개인 설정 - 예제 (p. 341)
- 콘텐츠 기반 동적 오리진 선택 - 예제 (p. 345)
- 오류 상태 업데이트 - 예제 (p. 352)
- 요청 본문에 액세스 - 예 (p. 354)

일반 예제

이 섹션의 예제는 CloudFront에서 Lambda@Edge를 사용하는 몇 가지 일반적인 방법을 보여줍니다.

주제

- 예: A/B 테스트 (p. 330)
- 예제: 응답 헤더 재정의 (p. 332)

예: A/B 테스트

다음 예제를 사용하여 리디렉션을 만들거나 URL을 변경하지 않고 두 가지 다른 버전의 이미지를 테스트할 수 있습니다. 이 예제는 최종 사용자 요청의 쿠키를 읽고 그에 따라 요청 URL을 수정합니다. 최종 사용자가 예상 값 중 하나와 함께 쿠키를 보내지 않는 경우, 이 예제에서는 최종 사용자를 URL 중 하나에 임의로 할당합니다.

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;
    const headers = request.headers;

    if (request.uri !== '/experiment-pixel.jpg') {
        // do not process if this is not an A-B test request
        callback(null, request);
        return;
    }

    const cookieExperimentA = 'X-Experiment-Name=A';
```

```
    const cookieExperimentB = 'X-Experiment-Name=B';
    const pathExperimentA = '/experiment-group/control-pixel.jpg';
    const pathExperimentB = '/experiment-group/treatment-pixel.jpg';

    /*
     * Lambda at the Edge headers are array objects.
     *
     * Client may send multiple Cookie headers, i.e.:
     * > GET /viewerRes/test HTTP/1.1
     * > User-Agent: curl/7.18.1 (x86_64-unknown-linux-gnu) libcurl/7.18.1
 OpenSSL/1.0.1u zlib/1.2.3
     * > Cookie: First=1; Second=2
     * > Cookie: ClientCode=abc
     * > Host: example.com
     *
     * You can access the first Cookie header at headers["cookie"][0].value
     * and the second at headers["cookie"][1].value.
     *
     * Header values are not parsed. In the example above,
     * headers["cookie"][0].value is equal to "First=1; Second=2"
     */
    let experimentUri;
    if (headers.cookie) {
        for (let i = 0; i < headers.cookie.length; i++) {
            if (headers.cookie[i].value.indexOf(cookieExperimentA) >= 0) {
                console.log('Experiment A cookie found');
                experimentUri = pathExperimentA;
                break;
            } else if (headers.cookie[i].value.indexOf(cookieExperimentB) >= 0) {
                console.log('Experiment B cookie found');
                experimentUri = pathExperimentB;
                break;
            }
        }
    }

    if (!experimentUri) {
        console.log('Experiment cookie has not been found. Throwing dice...');
        if (Math.random() < 0.75) {
            experimentUri = pathExperimentA;
        } else {
            experimentUri = pathExperimentB;
        }
    }

    request.uri = experimentUri;
    console.log(`Request uri set to "${request.uri}"`);
    callback(null, request);
};
```

Python

```
 import json
 import random

 def lambda_handler(event, context):
     request = event['Records'][0]['cf']['request']
     headers = request['headers']

     if request['uri'] != '/experiment-pixel.jpg':
         # Not an A/B Test
         return request

     cookieExperimentA, cookieExperimentB = 'X-Experiment-Name=A', 'X-Experiment-
Name=B'
```

```
    pathExperimentA, pathExperimentB = '/experiment-group/control-pixel.jpg', '/
experiment-group/treatment-pixel.jpg'

    '''
    Lambda at the Edge headers are array objects.

    Client may send multiple cookie headers. For example:
    > GET /viewerRes/test HTTP/1.1
    > User-Agent: curl/7.18.1 (x86_64-unknown-linux-gnu) libcurl/7.18.1 OpenSSL/1.0.1u
 zlib/1.2.3
    > Cookie: First=1; Second=2
    > Cookie: ClientCode=abc
    > Host: example.com

    You can access the first Cookie header at headers["cookie"][0].value
    and the second at headers["cookie"][1].value.

    Header values are not parsed. In the example above,
    headers["cookie"][0].value is equal to "First=1; Second=2"
    '''

    experimentUri = ""

    for cookie in headers.get('cookie', []):
        if cookieExperimentA in cookie['value']:
            print("Experiment A cookie found")
            experimentUri = pathExperimentA
            break
        elif cookieExperimentB in cookie['value']:
            print("Experiment B cookie found")
            experimentUri = pathExperimentB
            break

    if not experimentUri:
        print("Experiment cookie has not been found. Throwing dice...")
        if random.random() < 0.75:
            experimentUri = pathExperimentA
        else:
            experimentUri = pathExperimentB

    request['uri'] = experimentUri
    print(f"Request uri set to {experimentUri}")
    return request
```

예제: 응답 헤더 재정의

다음 예제는 다른 헤더의 값을 기준으로 응답 헤더의 값을 변경하는 방법을 보여줍니다.

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
   const response = event.Records[0].cf.response;
   const headers = response.headers;

   const headerNameSrc = 'X-Amz-Meta-Last-Modified';
   const headerNameDst = 'Last-Modified';

   if (headers[headerNameSrc.toLowerCase()]) {
      headers[headerNameDst.toLowerCase()] = [
         headers[headerNameSrc.toLowerCase()][0],
      ];
```

```
      console.log(`Response header "${headerNameDst}" was set to ` +
                `"${headers[headerNameDst.toLowerCase()][0].value}"`);
   }

   callback(null, response);
};
```

Python

```
import json

def lambda_handler(event, context):
    response = event["Records"][0]["cf"]["response"]
    headers = response["headers"]

    headerNameSrc = "X-Amz-Meta-Last-Modified"
    headerNameDst = "Last-Modified"

    if headers.get(headerNameSrc.lower(), None):
        headers[headerNameDst.lower()] = [headers[headerNameSrc.lower()][0]]
        print(f"Response header {headerNameDst.lower()} was set to
{headers[headerNameSrc.lower()][0]}")

    return response
```

응답 생성 - 예제

이 섹션의 예제는 Lambda@Edge를 사용하여 응답을 생성하는 방법을 보여줍니다.

주제

- 예제: 정적 콘텐츠 서비스(생성된 응답) (p. 333)
- 예제: Gzip 압축 콘텐츠(생성된 응답)로 정적 웹 사이트 콘텐츠 서비스 (p. 335)
- 예제: HTTP 리디렉션 생성(생성된 응답) (p. 336)

예제: 정적 콘텐츠 서비스(생성된 응답)

다음 예제는 Lambda 함수를 사용하여 정적 웹사이트 콘텐츠를 서비스하는 방법을 보여줍니다. 이는 오리진 서버에 대한 로드를 줄이고 전체 지연 시간을 단축합니다.

> **Note**
>
> 최종 사용자 요청 및 오리진 요청 이벤트에 대해 HTTP 응답을 만들 수 있습니다. 자세한 내용은 요청 트리거에서 HTTP 응답 생성 (p. 326) 단원을 참조하십시오. 오리진 및 최종 사용자 응답 이벤트에서 HTTP 응답을 바꿀 수도 있습니다. 자세한 내용은 오리진 응답 트리거에서 HTTP 응답 업데이트 (p. 329) 단원을 참조하십시오.

Node.js

```
'use strict';

const content = `
<\!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Simple Lambda@Edge Static Content Response</title>
  </head>
```

```
  <body>
    <p>Hello from Lambda@Edge!</p>
  </body>
</html>
`;

exports.handler = (event, context, callback) => {
    /*
     * Generate HTTP OK response using 200 status code with HTML body.
     */
    const response = {
        status: '200',
        statusDescription: 'OK',
        headers: {
            'cache-control': [{
                key: 'Cache-Control',
                value: 'max-age=100'
            }],
            'content-type': [{
                key: 'Content-Type',
                value: 'text/html'
            }],
            'content-encoding': [{
                key: 'Content-Encoding',
                value: 'UTF-8'
            }],
        },
        body: content,
    };
    callback(null, response);
};
```

Python

```
import json

CONTENT = """
<\!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Simple Lambda@Edge Static Content Response</title>
  </head>
  <body>
    <p>Hello from Lambda@Edge!</p>
  </body>
</html>
"""

def lambda_handler(event, context):
    # Generate HTTP OK response using 200 status code with HTML body.
    response = {
        'status': '200',
        'statusDescription': 'OK',
        'headers': {
            'cache-control': [
                {
                    'key': 'Cache-Control',
                    'value': 'max-age=100'
                }
            ],
            "content-type": [
                {
                    'key': 'Content-Type',
                    'value': 'text/html'
```

```
                }
            ],
            'content-encoding': [
                {
                    'key': 'Content-Encoding',
                    'value': 'UTF-8'
                }
            ]
        },
        'body': CONTENT
    }
    return response
```

예제: Gzip 압축 콘텐츠(생성된 응답)로 정적 웹 사이트 콘텐츠 서비스

이 함수는 Lambda 함수를 사용하여 gzip 압축 콘텐츠로 정적 웹 사이트 콘텐츠를 서비스하는 방법을 보여줍니다. 이는 오리진 서버에 대한 로드를 줄이고 전체 지연 시간을 단축합니다.

최종 사용자 요청 및 오리진 요청 이벤트에 대해 HTTP 응답을 만들 수 있습니다. 자세한 내용은 요청 트리거에서 HTTP 응답 생성 (p. 326) 단원을 참조하십시오.

Node.js

```
'use strict';

const zlib = require('zlib');

const content = `
<\!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Simple Lambda@Edge Static Content Response</title>
  </head>
  <body>
    <p>Hello from Lambda@Edge!</p>
  </body>
</html>
`;

exports.handler = (event, context, callback) => {

    /*
     * Generate HTTP OK response using 200 status code with a gzip compressed content
 HTML body.
     */

    const buffer = zlib.gzipSync(content);
    const base64EncodedBody = buffer.toString('base64');

    var response = {
        headers: {
            'content-type': [{key:'Content-Type', value: 'text/html; charset=utf-8'}],
            'content-encoding' : [{key:'Content-Encoding', value: 'gzip'}]
         },
        body: base64EncodedBody,
        bodyEncoding: 'base64',
        status: '200',
        statusDescription: "OK"
     }
```

```
    callback(null, response);
};
```

Python

```
import json
import zlib
import base64

CONTENT = """
<\!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Simple Lambda@Edge Static Content Response</title>
  </head>
  <body>
    <p>Hello from Lambda@Edge!</p>
  </body>
</html>
"""

def lambda_handler(event, context):
    # Generate HTTP OK response using 200 status code with a gzip compressed content
HTML body
    buf = zlib.compress(CONTENT.encode('utf-8'))
    base64EncodedBody = base64.b64encode(buf).decode('utf-8')
    response = {
        'headers': {
            'content-type': [
                {
                    'key': 'Content-Type',
                    'value': 'text/html; charset=utf-8'
                }
            ],
            'content-encoding': [
                {
                    'key': 'Content-Encoding',
                    'value': 'gzip'
                }
            ]
        },
        'body': base64EncodedBody,
        'bodyEncoding': 'base64',
        'status': '200',
        'statusDescription': 'OK'
    }
    return response
```

예제: HTTP 리디렉션 생성(생성된 응답)

다음 예제는 HTTP 리디렉션을 만드는 방법을 보여줍니다.

> **Note**
>
> 최종 사용자 요청 및 오리진 요청 이벤트에 대해 HTTP 응답을 만들 수 있습니다. 자세한 내용은 요청 트리거에서 HTTP 응답 생성 (p. 326) 단원을 참조하십시오.

Node.js

```
'use strict';
```

```
exports.handler = (event, context, callback) => {
    /*
     * Generate HTTP redirect response with 302 status code and Location header.
     */
    const response = {
        status: '302',
        statusDescription: 'Found',
        headers: {
            location: [{
                key: 'Location',
                value: 'http://docs.aws.amazon.com/lambda/latest/dg/lambda-edge.html',
            }],
        },
    };
    callback(null, response);
};
```

Python

```
 def lambda_handler(event, context):

     # Generate HTTP redirect response with 302 status code and Location header.

     response = {
         'status': '302',
         'statusDescription': 'Found',
         'headers': {
             'location': [{
                 'key': 'Location',
                 'value': 'http://docs.aws.amazon.com/lambda/latest/dg/lambda-
edge.html'
             }]
         }
     }

     return response
```

쿼리 문자열 처리 - 예제

이 섹션의 예제에는 Lambda@Edge를 쿼리 문자열과 함께 사용할 수 있는 방법이 포함되어 있습니다.

주제

- 예: 쿼리 문자열 파라미터를 기반으로 헤더 추가 (p. 337)
- 예: 쿼리 문자열 파라미터를 정규화하여 캐시 적중률 향상 (p. 338)
- 예: 인증되지 않은 사용자를 로그인 페이지로 리디렉션 (p. 340)

예: 쿼리 문자열 파라미터를 기반으로 헤더 추가

다음 예제에서는 쿼리 문자열 파라미터의 키-값 페어를 가져오고 그 값을 토대로 헤더를 추가하는 방법을 보여줍니다.

Node.js

```
'use strict';

const querystring = require('querystring');
exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;
```

```
    /* When a request contains a query string key-value pair but the origin server
     * expects the value in a header, you can use this Lambda function to
     * convert the key-value pair to a header. Here's what the function does:
     * 1. Parses the query string and gets the key-value pair.
     * 2. Adds a header to the request using the key-value pair that the function got
 in step 1.
     */

    /* Parse request querystring to get javascript object */
    const params = querystring.parse(request.querystring);

    /* Move auth param from querystring to headers */
    const headerName = 'Auth-Header';
    request.headers[headerName.toLowerCase()] = [{ key: headerName, value:
 params.auth }];
    delete params.auth;

    /* Update request querystring */
    request.querystring = querystring.stringify(params);

    callback(null, request);
}

;
```

Python

```
from urllib.parse import parse_qs, urlencode

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']

    '''
    When a request contains a query string key-value pair but the origin server
    expects the value in a header, you can use this Lambda function to
    convert the key-value pair to a header. Here's what the function does:
        1. Parses the query string and gets the key-value pair.
        2. Adds a header to the request using the key-value pair that the function got
 in step 1.
    '''

    # Parse request querystring to get dictionary/json
    params = {k : v[0] for k, v in parse_qs(request['querystring']).items()}

    # Move auth param from querystring to headers
    headerName = 'Auth-Header'
    request['headers'][headerName.lower()] = [{'key': headerName, 'value':
 params['auth']}]
    del params['auth']

    # Update request querystring
    request['querystring'] = urlencode(params)

    return request
```

예: 쿼리 문자열 파라미터를 정규화하여 캐시 적중률 향상

다음 예제에서는 CloudFront가 요청을 오리진에 전달하기 전에 쿼리 문자열을 다음과 같이 변경하여 캐시 적중률을 향상하는 방법을 보여 줍니다.

- 파라미터 이름을 기준으로 키-값 페어를 알파벳순으로 정렬

- 키-값 페어의 대/소문자를 소문자로 변경

자세한 내용은 쿼리 문자열 파라미터 기반의 콘텐츠 캐싱 (p. 196) 단원을 참조하십시오.

Node.js

```
'use strict';

const querystring = require('querystring');

exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;
    /* When you configure a distribution to forward query strings to the origin and
     * to cache based on a whitelist of query string parameters, we recommend
     * the following to improve the cache-hit ratio:
     * - Always list parameters in the same order.
     * - Use the same case for parameter names and values.
     *
     * This function normalizes query strings so that parameter names and values
     * are lowercase and parameter names are in alphabetical order.
     *
     * For more information, see:
     * http://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/
QueryStringParameters.html
     */

    console.log('Query String: ', request.querystring);

    /* Parse request query string to get javascript object */
    const params = querystring.parse(request.querystring.toLowerCase());
    const sortedParams = {};

    /* Sort param keys */
    Object.keys(params).sort().forEach(key => {
        sortedParams[key] = params[key];
    });

    /* Update request querystring with normalized  */
    request.querystring = querystring.stringify(sortedParams);

    callback(null, request);
};
```

Python

```
 from urllib.parse import parse_qs, urlencode

 def lambda_handler(event, context):
     request = event['Records'][0]['cf']['request']
     '''
     When you configure a distribution to forward query strings to the origin and
     to cache based on a whitelist of query string parameters, we recommend
     the following to improve the cache-hit ratio:
     Always list parameters in the same order.
     - Use the same case for parameter names and values.

     This function normalizes query strings so that parameter names and values
     are lowercase and parameter names are in alphabetical order.

     For more information, see:
     http://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/
QueryStringParameters.html
     '''
```

```
    print("Query string: ", request["querystring"])

    # Parse request query string to get js object
    params = {k : v[0] for k, v in parse_qs(request['querystring'].lower()).items()}

    # Sort param keys
    sortedParams = sorted(params.items(), key=lambda x: x[0])

    # Update request querystring with normalized
    request['querystring'] = urlencode(sortedParams)

    return request
```

예: 인증되지 않은 사용자를 로그인 페이지로 리디렉션

다음 예제에서는 사용자가 자격 증명을 입력하지 않은 경우 사용자를 로그인 페이지로 리디렉션하는 방법을 보여 줍니다.

Node.js

```
'use strict';

function parseCookies(headers) {
    const parsedCookie = {};
    if (headers.cookie) {
        headers.cookie[0].value.split(';').forEach((cookie) => {
            if (cookie) {
                const parts = cookie.split('=');
                parsedCookie[parts[0].trim()] = parts[1].trim();
            }
        });
    }
    return parsedCookie;
}

exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;
    const headers = request.headers;

    /* Check for session-id in request cookie in viewer-request event,
     * if session-id is absent, redirect the user to sign in page with original
     * request sent as redirect_url in query params.
     */

    /* Check for session-id in cookie, if present then proceed with request */
    const parsedCookies = parseCookies(headers);
    if (parsedCookies && parsedCookies['session-id']) {
        callback(null, request);
        return;
    }

    /* URI encode the original request to be sent as redirect_url in query params */
    const encodedRedirectUrl = encodeURIComponent(`https://
${headers.host[0].value}${request.uri}?${request.querystring}`);
    const response = {
        status: '302',
        statusDescription: 'Found',
        headers: {
            location: [{
                key: 'Location',
                value: `https://www.example.com/signin?redirect_url=
${encodedRedirectUrl}`,
            }],
```

```
        },
    };
    callback(null, response);
};
```

Python

```
import urllib

def parseCookies(headers):
    parsedCookie = {}
    if headers.get('cookie'):
        for cookie in headers['cookie'][0]['value'].split(';'):
            if cookie:
                parts = cookie.split('=')
                parsedCookie[parts[0].strip()] = parts[1].strip()
    return parsedCookie

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    headers = request['headers']

    '''
    Check for session-id in request cookie in viewer-request event,
    if session-id is absent, redirect the user to sign in page with original
    request sent as redirect_url in query params.
    '''

    # Check for session-id in cookie, if present, then proceed with request
    parsedCookies = parseCookies(headers)

    if parsedCookies and parsedCookies['session-id']:
        return request

    # URI encode the original request to be sent as redirect_url in query params
    redirectUrl = "https://%s%s?%s" % (headers['host'][0]['value'], request['uri'],
 request['querystring'])
    encodedRedirectUrl = urllib.parse.quote_plus(redirectUrl.encode('utf-8'))

    response = {
        'status': '302',
        'statusDescription': 'Found',
        'headers': {
            'location': [{
                'key': 'Location',
                'value': 'https://www.example.com/signin?redirect_url=%s' %
 encodedRedirectUrl
            }]
        }
    }
    return response
```

국가 또는 디바이스 유형 헤더에 따른 콘텐츠 개인 설정 - 예제

이 섹션의 예제는 Lambda@Edge를 사용하여 뷰어가 사용하는 위치나 장치 유형에 따라 동작을 사용자 정의하는 방법을 보여줍니다.

주제

- 예: 최종 사용자 요청을 국가별 URL로 리디렉션 (p. 342)

- 예: 디바이스를 기반으로 객체의 다양한 버전 서비스 (p. 343)

예: 최종 사용자 요청을 국가별 URL로 리디렉션

다음 예제에서는 국가별 URL이 포함된 HTTP 리디렉션 응답을 생성하고 해당 응답을 최종 사용자에게 반환하는 방법을 보여 줍니다. 이 작업은 국가별 응답을 제공하려는 경우에 유용합니다. 예:

- 국가별 하위 도메인이 있는 경우(예: us.example.com 및 tw.example.com) 최종 사용자가 example.com을 요청할 때 리디렉션 응답을 생성할 수 있습니다.
- 비디오를 스트리밍하지만 특정 국가에서 콘텐츠를 스트리밍할 권한이 없는 경우 비디오를 볼 수 없는 이유를 설명하는 페이지로 해당 국가의 사용자를 리디렉션할 수 있습니다.

다음을 참조하십시오.

- `CloudFront-Viewer-Country` 헤더 값을 기반으로 캐시하도록 배포를 구성해야 합니다. 자세한 내용은 선택한 요청 헤더 기반의 캐시 (p. 46) 단원을 참조하십시오.
- CloudFront는 최종 사용자 요청 이벤트 뒤에 `CloudFront-Viewer-Country` 헤더를 추가합니다. 이 예제를 사용하려면 오리진 요청 이벤트에 대한 트리거를 생성해야 합니다.

Node.js

```
'use strict';

/* This is an origin request function */
exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;
    const headers = request.headers;

    /*
     * Based on the value of the CloudFront-Viewer-Country header, generate an
     * HTTP status code 302 (Redirect) response, and return a country-specific
     * URL in the Location header.
     * NOTE: 1. You must configure your distribution to cache based on the
     *          CloudFront-Viewer-Country header. For more information, see
     *          http://docs.aws.amazon.com/console/cloudfront/cache-on-selected-headers
     *       2. CloudFront adds the CloudFront-Viewer-Country header after the viewer
     *          request event. To use this example, you must create a trigger for the
     *          origin request event.
     */

    let url = 'https://example.com/';
    if (headers['cloudfront-viewer-country']) {
        const countryCode = headers['cloudfront-viewer-country'][0].value;
        if (countryCode === 'TW') {
            url = 'https://tw.example.com/';
        } else if (countryCode === 'US') {
            url = 'https://us.example.com/';
        }
    }

    const response = {
        status: '302',
        statusDescription: 'Found',
        headers: {
            location: [{
                key: 'Location',
                value: url,
            }],
        },
```

```
    };
    callback(null, response);
};
```

Python

```
# This is an origin request function

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    headers = request['headers']

    '''
     Based on the value of the CloudFront-Viewer-Country header, generate an
     HTTP status code 302 (Redirect) response, and return a country-specific
     URL in the Location header.
     NOTE: 1. You must configure your distribution to cache based on the
              CloudFront-Viewer-Country header. For more information, see
              http://docs.aws.amazon.com/console/cloudfront/cache-on-selected-headers
           2. CloudFront adds the CloudFront-Viewer-Country header after the viewer
              request event. To use this example, you must create a trigger for the
              origin request event.
    '''

    url = 'https://example.com/'
    viewerCountry = headers.get('cloudfront-viewer-country')
    if viewerCountry:
        countryCode = viewerCountry[0]['value']
        if countryCode == 'TW':
            url = 'https://tw.example.com/'
        elif countryCode == 'US':
            url = 'https://us.example.com/'

    response = {
        'status': '302',
        'statusDescription': 'Found',
        'headers': {
            'location': [{
                'key': 'Location',
                'value': url
            }]
        }
    }

    return response
```

예: 디바이스를 기반으로 객체의 다양한 버전 서비스

다음 예제에서는 사용자가 사용 중인 디바이스 유형(예: 모바일 디바이스 또는 태블릿)을 기반으로 객체의 다양한 버전을 서비스하는 방법을 보여 줍니다. 다음을 참조하십시오.

- `CloudFront-Is-*-Viewer` 헤더 값을 기반으로 캐시하도록 배포를 구성해야 합니다. 자세한 내용은 선택한 요청 헤더 기반의 캐시 (p. 46) 단원을 참조하십시오.
- CloudFront는 최종 사용자 요청 이벤트 뒤에 `CloudFront-Is-*-Viewer` 헤더를 추가합니다. 이 예제를 사용하려면 오리진 요청 이벤트에 대한 트리거를 생성해야 합니다.

Node.js

```
'use strict';
```

```
/* This is an origin request function */
exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;
    const headers = request.headers;

    /*
     * Serve different versions of an object based on the device type.
     * NOTE: 1. You must configure your distribution to cache based on the
     *          CloudFront-Is-*-Viewer headers. For more information, see
     *          the following documentation:
     *          http://docs.aws.amazon.com/console/cloudfront/cache-on-selected-headers
     *          http://docs.aws.amazon.com/console/cloudfront/cache-on-device-type
     *       2. CloudFront adds the CloudFront-Is-*-Viewer headers after the viewer
     *          request event. To use this example, you must create a trigger for the
     *          origin request event.
     */

    const desktopPath = '/desktop';
    const mobilePath = '/mobile';
    const tabletPath = '/tablet';
    const smarttvPath = '/smarttv';

    if (headers['cloudfront-is-desktop-viewer']
        && headers['cloudfront-is-desktop-viewer'][0].value === 'true') {
        request.uri = desktopPath + request.uri;
    } else if (headers['cloudfront-is-mobile-viewer']
                && headers['cloudfront-is-mobile-viewer'][0].value === 'true') {
        request.uri = mobilePath + request.uri;
    } else if (headers['cloudfront-is-tablet-viewer']
                && headers['cloudfront-is-tablet-viewer'][0].value === 'true') {
        request.uri = tabletPath + request.uri;
    } else if (headers['cloudfront-is-smarttv-viewer']
                && headers['cloudfront-is-smarttv-viewer'][0].value === 'true') {
        request.uri = smarttvPath + request.uri;
    }
    console.log(`Request uri set to "${request.uri}"`);

    callback(null, request);
};
```

Python

```
# This is an origin request function
 def lambda_handler(event, context):
     request = event['Records'][0]['cf']['request']
     headers = request['headers']

     '''
      Serve different versions of an object based on the device type.
      NOTE: 1. You must configure your distribution to cache based on the
               CloudFront-Is-*-Viewer headers. For more information, see
               the following documentation:
               http://docs.aws.amazon.com/console/cloudfront/cache-on-selected-headers
               http://docs.aws.amazon.com/console/cloudfront/cache-on-device-type
            2. CloudFront adds the CloudFront-Is-*-Viewer headers after the viewer
               request event. To use this example, you must create a trigger for the
               origin request event.
     '''

     desktopPath = '/desktop';
     mobilePath = '/mobile';
     tabletPath = '/tablet';
     smarttvPath = '/smarttv';
```

```
    if 'cloudfront-is-desktop-viewer' in headers and headers['cloudfront-is-desktop-
viewer'][0]['value'] == 'true':
        request['uri'] = desktopPath + request['uri']
    elif 'cloudfront-is-mobile-viewer' in headers and headers['cloudfront-is-mobile-
viewer'][0]['value'] == 'true':
        request['uri'] = mobilePath + request['uri']
    elif 'cloudfront-is-tablet-viewer' in headers and headers['cloudfront-is-tablet-
viewer'][0]['value'] == 'true':
        request['uri'] = tabletPath + request['uri']
    elif 'cloudfront-is-smarttv-viewer' in headers and headers['cloudfront-is-smarttv-
viewer'][0]['value'] == 'true':
        request['uri'] = smarttvPath + request['uri']

    print("Request uri set to %s" % request['uri'])

    return request
```

콘텐츠 기반 동적 오리진 선택 - 예제

이 섹션의 예제는 Lambda@Edge를 사용하여 요청 정보를 기반으로 다른 오리진으로 라우팅하는 방법을 보여줍니다.

주제

- 예제: 오리진 요청 트리거를 사용하여 사용자 지정 오리진에서 Amazon S3 오리진으로 변경 (p. 345)
- 예제: 오리진 요청 트리거를 사용하여 Amazon S3 오리진 리전 변경 (p. 346)
- 예제: 오리진 요청 트리거를 사용하여 Amazon S3 오리진에서 사용자 지정 오리진으로 변경 (p. 349)
- 예제: 오리진 요청 트리거를 사용하여 하나의 Amazon S3 버킷에서 다른 버킷으로 점진적으로 트래픽 전송 (p. 350)
- 예제: 오리진 요청 트리거를 사용하여 국가 헤더를 기반으로 한 오리진 도메인 이름 변경 (p. 351)

예제: 오리진 요청 트리거를 사용하여 사용자 지정 오리진에서 Amazon S3 오리진으로 변경

이 함수는 오리진 요청 트리거를 사용하여 사용자 지정 오리진에서 콘텐츠를 요청 속성을 기반으로 가져오는 Amazon S3 오리진으로 변경하는 방법을 보여줍니다.

Node.js

```
'use strict';

 const querystring = require('querystring');

 exports.handler = (event, context, callback) => {
     const request = event.Records[0].cf.request;

     /**
      * Reads query string to check if S3 origin should be used, and
      * if true, sets S3 origin properties.
      */

     const params = querystring.parse(request.querystring);

     if (params['useS3Origin']) {
         if (params['useS3Origin'] === 'true') {
             const s3DomainName = 'my-bucket.s3.amazonaws.com';
```

```
            /* Set S3 origin fields */
            request.origin = {
                s3: {
                    domainName: s3DomainName,
                    region: '',
                    authMethod: 'none',
                    path: '',
                    customHeaders: {}
                }
            };
            request.headers['host'] = [{ key: 'host', value: s3DomainName}];
        }
    }

    callback(null, request);
};
```

Python

```
from urllib.parse import parse_qs

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    '''
    Reads query string to check if S3 origin should be used, and
    if true, sets S3 origin properties
    '''
    params = {k: v[0] for k, v in parse_qs(request['querystring']).items()}
    if params.get('useS3Origin') == 'true':
        s3DomainName = 'my-bucket.s3.amazonaws.com'

        # Set S3 origin fields
        request['origin'] = {
            's3': {
                'domainName': s3DomainName,
                'region': '',
                'authMethod': 'none',
                'path': '',
                'customHeaders': {}
            }
        }
        request['headers']['host'] = [{'key': 'host', 'value': s3DomainName}]
    return request
```

예제: 오리진 요청 트리거를 사용하여 Amazon S3 오리진 리전 변경

이 함수는 오리진 요청 트리거를 사용하여 콘텐츠를 요청 속성을 기반으로 가져오는 Amazon S3 오리진을 변경하는 방법을 보여줍니다.

이 예제에서는 CloudFront-Viewer-Country 헤더의 값을 사용하여 S3 버킷 도메인 이름을 최종 사용자와 인접한 리전에 있는 버킷으로 업데이트합니다. 이는 여러 방법에서 유용할 수 있습니다.

- 지정된 리전이 최종 사용자의 국가와 더욱 근접할 때 지연 시간을 감소합니다.
- 데이터가 요청이 온 곳과 동일한 국가에 있는 리전에서 서비스되는지 확인함으로써 데이터 주권을 제공합니다.

이 예제를 활용하려면 다음과 같이 해야 합니다.

- CloudFront-Viewer-Country 헤더를 기반으로 캐시하도록 배포를 구성합니다. 자세한 내용은 선택한 요청 헤더 기반의 캐시 (p. 46) 단원을 참조하십시오.
- 오리진 요청 이벤트에서 이 함수에 대한 트리거를 생성합니다. CloudFront는 최종 사용자 요청 이벤트 후에 CloudFront-Viewer-Country 헤더를 추가합니다. 이 예제를 사용하려면 오리진 요청에서 함수가 실행되는지 확인해야 합니다.

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;

    /**
     * This blueprint demonstrates how an origin-request trigger can be used to
     * change the origin from which the content is fetched, based on request
 properties.
     * In this example, we use the value of the CloudFront-Viewer-Country header
     * to update the S3 bucket domain name to a bucket in a Region that is closer to
     * the viewer.
     *
     * This can be useful in several ways:
     *      1) Reduces latencies when the Region specified is nearer to the viewer's
     *         country.
     *      2) Provides data sovereignty by making sure that data is served from an
     *         origin that's in the same country that the request came from.
     *
     * NOTE: 1. You must configure your distribution to cache based on the
     *          CloudFront-Viewer-Country header. For more information, see
     *          http://docs.aws.amazon.com/console/cloudfront/cache-on-selected-headers
     *       2. CloudFront adds the CloudFront-Viewer-Country header after the viewer
     *          request event. To use this example, you must create a trigger for the
     *          origin request event.
     */

    const countryToRegion = {
        'DE': 'eu-central-1',
        'IE': 'eu-west-1',
        'GB': 'eu-west-2',
        'FR': 'eu-west-3',
        'JP': 'ap-northeast-1',
        'IN': 'ap-south-1'
    };

    if (request.headers['cloudfront-viewer-country']) {
        const countryCode = request.headers['cloudfront-viewer-country'][0].value;
        const region = countryToRegion[countryCode];

        /**
         * If the viewer's country is not in the list you specify, the request
         * goes to the default S3 bucket you've configured.
         */
        if (region) {
            /**
             * If you've set up OAI, the bucket policy in the destination bucket
             * should allow the OAI GetObject operation, as configured by default
             * for an S3 origin with OAI. Another requirement with OAI is to provide
             * the Region so it can be used for the SIGV4 signature. Otherwise, the
             * Region is not required.
             */
            request.origin.s3.region = region;
            const domainName = `my-bucket-in-${region}.s3.amazonaws.com`;
            request.origin.s3.domainName = domainName;
```

```
            request.headers['host'] = [{ key: 'host', value: domainName }];
        }
    }

    callback(null, request);
};
```

Python

```
def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']

    '''
    This blueprint demonstrates how an origin-request trigger can be used to
    change the origin from which the content is fetched, based on request properties.
    In this example, we use the value of the CloudFront-Viewer-Country header
    to update the S3 bucket domain name to a bucket in a Region that is closer to
    the viewer.

    This can be useful in several ways:
        1) Reduces latencies when the Region specified is nearer to the viewer's
             country.
        2) Provides data sovereignty by making sure that data is served from an
             origin that's in the same country that the request came from.

    NOTE: 1. You must configure your distribution to cache based on the
             CloudFront-Viewer-Country header. For more information, see
             http://docs.aws.amazon.com/console/cloudfront/cache-on-selected-headers
          2. CloudFront adds the CloudFront-Viewer-Country header after the viewer
             request event. To use this example, you must create a trigger for the
             origin request event.
    '''

    countryToRegion = {
        'DE': 'eu-central-1',
        'IE': 'eu-west-1',
        'GB': 'eu-west-2',
        'FR': 'eu-west-3',
        'JP': 'ap-northeast-1',
        'IN': 'ap-south-1'
    }

    viewerCountry = request['headers'].get('cloudfront-viewer-country')
    if viewerCountry:
        countryCode = viewerCountry[0]['value']
        region = countryToRegion.get(countryCode)

        # If the viewer's country in not in the list you specify, the request
        # goes to the default S3 bucket you've configured
        if region:
            '''
            If you've set up OAI, the bucket policy in the destination bucket
            should allow the OAI GetObject operation, as configured by default
            for an S3 origin with OAI. Another requirement with OAI is to provide
            the Region so it can be used for the SIGV4 signature. Otherwise, the
            Region is not required.
            '''
            request['origin']['s3']['region'] = region
            domainName = 'my-bucket-in-%s.s3.amazonaws.com' % region
            request['origin']['s3']['domainName'] = domainName
            request['headers']['host'] = [{'key': 'host', 'value': domainName}]

    return request
```

예제: 오리진 요청 트리거를 사용하여 Amazon S3 오리진에서 사용자 지정 오리진으로 변경

이 함수는 오리진 요청 트리거를 사용하여 콘텐츠를 요청 속성을 기반으로 가져오는 사용자 지정 오리진을 변경하는 방법을 보여줍니다.

Node.js

```
'use strict';

const querystring = require('querystring');

 exports.handler = (event, context, callback) => {
     const request = event.Records[0].cf.request;

     /**
      * Reads query string to check if custom origin should be used, and
      * if true, sets custom origin properties.
      */

     const params = querystring.parse(request.querystring);

     if (params['useCustomOrigin']) {
         if (params['useCustomOrigin'] === 'true') {

             /* Set custom origin fields*/
             request.origin = {
                 custom: {
                     domainName: 'www.example.com',
                     port: 443,
                     protocol: 'https',
                     path: '',
                     sslProtocols: ['TLSv1', 'TLSv1.1'],
                     readTimeout: 5,
                     keepaliveTimeout: 5,
                     customHeaders: {}
                 }
             };
             request.headers['host'] = [{ key: 'host', value: 'www.example.com'}];
         }
     }
    callback(null, request);
};
```

Python

```
from urllib.parse import parse_qs

 def lambda_handler(event, context):
     request = event['Records'][0]['cf']['request']

     # Reads query string to check if custom origin should be used, and
     # if true, sets custom origin properties

     params = {k: v[0] for k, v in parse_qs(request['querystring']).items()}

     if params.get('useCustomOrigin') == 'true':
             # Set custom origin fields
             request['origin'] = {
                 'custom': {
                     'domainName': 'www.example.com',
                     'port': 443,
```

```
                    'protocol': 'https',
                    'path': '',
                    'sslProtocols': ['TLSv1', 'TLSv1.1'],
                    'readTimeout': 5,
                    'keepaliveTimeout': 5,
                    'customHeaders': {}
                }
            }
            request['headers']['host'] = [{'key': 'host', 'value': 'www.example.com'}]

    return request
```

예제: 오리진 요청 트리거를 사용하여 하나의 Amazon S3 버킷에서 다른 버킷으로 점진적으로 트래픽 전송

이 함수는 통제된 방법을 통해 하나의 Amazon S3 버킷에서 다른 버킷으로 점진적으로 트래픽을 전송하는 방법을 보여줍니다.

Node.js

```
'use strict';

    function getRandomInt(min, max) {
        /* Random number is inclusive of min and max*/
        return Math.floor(Math.random() * (max - min + 1)) + min;
 }

exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;
    const BLUE_TRAFFIC_PERCENTAGE = 80;

    /**
      * This Lambda function demonstrates how to gradually transfer traffic from
      * one S3 bucket to another in a controlled way.
      * We define a variable BLUE_TRAFFIC_PERCENTAGE which can take values from
      * 1 to 100. If the generated randomNumber less than or equal to
 BLUE_TRAFFIC_PERCENTAGE, traffic
      * is re-directed to blue-bucket. If not, the default bucket that we've configured
      * is used.
      */

    const randomNumber = getRandomInt(1, 100);

if (randomNumber <= BLUE_TRAFFIC_PERCENTAGE) {
         const domainName = 'blue-bucket.s3.amazonaws.com';
         request.origin.s3.domainName = domainName;
         request.headers['host'] = [{ key: 'host', value: domainName}];
     }
    callback(null, request);
};
```

Python

```
 import math
 import random

 def getRandomInt(min, max):
     # Random number is inclusive of min and max
     return math.floor(random.random() * (max - min + 1)) + min

 def lambda_handler(min, max):
```

```
    request = event['Records'][0]['cf']['request']
    BLUE_TRAFFIC_PERCENTAGE = 80

    '''
    This Lambda function demonstrates how to gradually transfer traffic from
    one S3 bucket to another in a controlled way.
    We define a variable BLUE_TRAFFIC_PERCENTAGE which can take values from
    1 to 100. If the generated randomNumber less than or equal to
BLUE_TRAFFIC_PERCENTAGE, traffic
    is re-directed to blue-bucket. If not, the default bucket that we've configured
    is used.
    '''

    randomNumber = getRandomInt(1, 100)

    if randomNumber <= BLUE_TRAFFIC_PERCENTAGE:
        domainName = 'blue-bucket.s3.amazonaws.com'
        request['origin']['s3']['domainName'] = domainName
        request['heaaders']['host'] = [{'key': 'host', 'value': domainName}]

    return request
```

예제: 오리진 요청 트리거를 사용하여 국가 헤더를 기반으로 한 오리진 도메인 이름 변경

이 함수는 콘텐츠가 최종 사용자의 국가에서 가까운 오리진에서 서비스되도록 CloudFront-Viewer-Country 헤더를 기반으로 오리진 도메인 이름을 변경하는 방법을 보여줍니다.

배포에서 이 기능을 구현하면 다음과 같은 이점을 누릴 수 있습니다.

- 지정된 리전이 최종 사용자의 국가와 더욱 근접할 때 지연 시간이 감소
- 데이터가 요청이 온 곳과 동일한 국가에 있는 오리진에서 서비스되는지 확인함으로써 데이터 주권을 제공

이 기능을 활성화하려면 CloudFront-Viewer-Country 헤더를 기반으로 캐시할 배포를 구성해야 합니다. 자세한 내용은 선택한 요청 헤더 기반의 캐시 (p. 46) 단원을 참조하십시오.

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
     const request = event.Records[0].cf.request;

  if (request.headers['cloudfront-viewer-country']) {
         const countryCode = request.headers['cloudfront-viewer-country'][0].value;
         if (countryCode === 'UK' || countryCode === 'DE' || countryCode === 'IE' ) {
             const domainName = 'eu.example.com';
             request.origin.custom.domainName = domainName;
             request.headers['host'] = [{key: 'host', value: domainName}];
         }
     }

    callback(null, request);
};
```

Python

```
 def lambda_handler(event, context):
     request = event['Records'][0]['cf']['request']
```

```
    viwerCountry = request['headers'].get('cloudfront-viewer-country')
    if viewerCountry:
        countryCode = viewerCountry[0]['value']
        if countryCode == 'UK' or countryCode == 'DE' or countryCode == 'IE':
            domainName = 'eu.example.com'
            request['origin']['custom']['domainName'] = domainName
            request['headers']['host'] = [{'key': 'host', 'value': domainName}]
    return request
```

오류 상태 업데이트 - 예제

이 섹션의 예제는 Lambda@Edge를 사용하여 사용자에게 반환되는 오류의 상태를 변경하는 방법에 대한 지침을 제공합니다.

주제

- 예제: 오리진 응답 트리거를 사용하여 오류 상태 코드를 200-OK로 업데이트 (p. 352)
- 예제: 오리진 응답 트리거를 사용하여 오류 상태 코드를 302-Found로 업데이트 (p. 353)

예제: 오리진 응답 트리거를 사용하여 오류 상태 코드를 200-OK로 업데이트

이 함수는 응답 상태를 200으로 업데이트하고 정적 본문 콘텐츠를 생성하여 다음 시나리오에서 최종 사용자에게 반환하는 방법을 보여줍니다.

- 함수가 오리진 응답에서 트리거됩니다.
- 오리진 서버의 응답 상태는 오류 상태 코드(4xx 또는 5xx)입니다.

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
    const response = event.Records[0].cf.response;

    /**
     * This function updates the response status to 200 and generates static
     * body content to return to the viewer in the following scenario:
     * 1. The function is triggered in an origin response
     * 2. The response status from the origin server is an error status code (4xx or
 5xx)
     */

    if (response.status >= 400 && response.status <= 599) {
        response.status = 200;
        response.statusDescription = 'OK';
        response.body = 'Body generation example';
    }

    callback(null, response);
};
```

Python

```
 def lambda_handler(event, context):
```

```
    response = event['Records'][0]['cf']['response']

    '''
    This function updates the response status to 200 and generates static
    body content to return to the viewer in the following scenario:
    1. The function is triggered in an origin response
    2. The response status from the origin server is an error status code (4xx or 5xx)
    '''

    if int(response['status']) >= 400 and int(response['status']) <= 599:
        response['status'] = 200
        response['statusDescription'] = 'OK'
        response['body'] = 'Body generation example'
    return response
```

예제: 오리진 응답 트리거를 사용하여 오류 상태 코드를 302-Found로 업데이트

이 함수는 HTTP 상태 코드를 302로 업데이트하여 다른 오리진이 구성된 다른 경로(캐시 동작)로 리디렉션하는 방법을 보여줍니다. 다음을 참조하십시오.

- 함수가 오리진 응답에서 트리거됩니다.
- 오리진 서버의 응답 상태는 오류 상태 코드(4xx 또는 5xx)입니다.

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
    const response = event.Records[0].cf.response;
    const request = event.Records[0].cf.request;

    /**
     * This function updates the HTTP status code in the response to 302, to redirect
 to another
     * path (cache behavior) that has a different origin configured. Note the
 following:
     * 1. The function is triggered in an origin response
     * 2. The response status from the origin server is an error status code (4xx or
 5xx)
     */

    if (response.status >= 400 && response.status <= 599) {
        const redirect_path = `/plan-b/path?${request.querystring}`;

        response.status = 302;
        response.statusDescription = 'Found';

        /* Drop the body, as it is not required for redirects */
        response.body = '';
        response.headers['location'] = [{ key: 'Location', value: redirect_path }];
    }

    callback(null, response);
};
```

Python

```
def lambda_handler(event, context):
```

```
    response = event['Records'][0]['cf']['response']
    request = event['Records'][0]['cf']['request']

    '''
    This function updates the HTTP status code in the response to 302, to redirect to
 another
    path (cache behavior) that has a different origin configured. Note the following:
    1. The function is triggered in an origin response
    2. The response status from the origin server is an error status code (4xx or 5xx)
    '''

    if int(response['status']) >= 400 and int(response['status']) <= 599:
        redirect_path = '/plan-b/path?%s' % request['querystring']

        response['status'] = 302
        response['statusDescription'] = 'Found'

        # Drop the body as it is not required for redirects
        response['body'] = ''
        response['headers']['location'] = [{'key': 'Location', 'value':
 redirect_path}]

    return response
```

요청 본문에 액세스 - 예

이 섹션의 예제는 Lambda@Edge를 사용하여 POST 요청을 처리하는 방법을 보여줍니다.

주제

- 예: 요청 트리거를 사용하여 HTML 양식 읽기 (p. 354)
- 예: 요청 트리거를 사용하여 HTML 양식 수정 (p. 356)

예: 요청 트리거를 사용하여 HTML 양식 읽기

이 기능은 "문의처" 양식 등과 같은 HTML 양식(웹 양식)으로 인해 생성된 POST 요청의 본문을 처리할 수 있는 방법을 보여줍니다. 예를 들어, 다음과 같은 HTML 양식이 있을 수 있습니다.

```
<html>
  <form action="http://example.com" method="post">
    Param 1: <input type="text" name="name1"><br>
    Param 2: <input type="text" name="name2"><br>
    input type="submit" value="Submit">
  </form>
</html>
```

예를 들어, 다음 함수는 CloudFront 최종 사용자 요청 또는 오리진 요청 시 실행되어야 합니다.

Node.js

```
'use strict';

const querystring = require('querystring');

/**
 * This function demonstrates how you can read the body of a POST request
 * generated by an HTML form (web form). The function is triggered in a
 * CloudFront viewer request or origin request event type.
 */
```

```
exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;

    if (request.method === 'POST') {
        /* HTTP body is always passed as base64-encoded string. Decode it. */
        const body = Buffer.from(request.body.data, 'base64').toString();

        /* HTML forms send the data in query string format. Parse it. */
        const params = querystring.parse(body);

        /* For demonstration purposes, we only log the form fields here.
         * You can put your custom logic here. For example, you can store the
         * fields in a database, such as AWS DynamoDB, and generate a response
         * right from your Lambda@Edge function.
         */
        for (let param in params) {
            console.log(`For "${param}" user submitted "${params[param]}".\n`);
        }
    }
    return callback(null, request);
};
```

Python

```
import base64
from urllib.parse import parse_qs

'''
Say there is a POST request body generated by an HTML such as:

<html>
  <form action="http://example.com" method="post">
    Param 1: <input type="text" name="name1"><br>
    Param 2: <input type="text" name="name2"><br>
    input type="submit" value="Submit">
  </form>
</html>

'''

'''
This function demonstrates how you can read the body of a POST request
generated by an HTML form (web form). The function is triggered in a
CloudFront viewer request or origin request event type.
'''

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']

    if request['method'] == 'POST':
        # HTTP body is always passed as base64-encoded string. Decode it
        body = base64.b64decode(request['body']['data'])

        # HTML forms send the data in query string format. Parse it
        params = {k: v[0] for k, v in parse_qs(body).items()}

        '''
        For demonstration purposes, we only log the form fields here.
        You can put your custom logic here. For example, you can store the
        fields in a database, such as AWS DynamoDB, and generate a response
        right from your Lambda@Edge function.
        '''
        for key, value in params.items():
            print("For %s use submitted %s" % (key, value))
```

```
    return request
```

예: 요청 트리거를 사용하여 HTML 양식 수정

이 기능은 "문의처" 양식 등과 같은 HTML 양식(웹 양식)으로 인해 생성된 POST 요청의 본문을 수정할 수 있는 방법을 보여줍니다. 이 함수는 CloudFront 최종 사용자 요청 또는 오리진 요청 시 실행됩니다.

Node.js

```
'use strict';

const querystring = require('querystring');

exports.handler = (event, context, callback) => {
    var request = event.Records[0].cf.request;
    if (request.method === 'POST') {
        /* Request body is being replaced. To do this, update the following
        /* three fields:
         *    1) body.action to 'replace'
         *    2) body.encoding to the encoding of the new data.
         *
         *       Set to one of the following values:
         *
         *           text - denotes that the generated body is in text format.
         *               Lambda@Edge will propagate this as is.
         *           base64 - denotes that the generated body is base64 encoded.
         *               Lambda@Edge will base64 decode the data before sending
         *               it to the origin.
         *    3) body.data to the new body.
         */
        request.body.action = 'replace';
        request.body.encoding = 'text';
        request.body.data = getUpdatedBody(request);
    }
    callback(null, request);
};

function getUpdatedBody(request) {
    /* HTTP body is always passed as base64-encoded string. Decode it. */
    const body = Buffer.from(request.body.data, 'base64').toString();

    /* HTML forms send data in query string format. Parse it. */
    const params = querystring.parse(body);

    /* For demonstration purposes, we're adding one more param.
     *
     * You can put your custom logic here. For example, you can truncate long
     * bodies from malicious requests.
     */
    params['new-param-name'] = 'new-param-value';
    return querystring.stringify(params);
}
```

Python

```
 import base64
 from urllib.parse import parse_qs, urlencode

 def lambda_handler(event, context):
     request = event['Records'][0]['cf']['request']
```

```
    if request['method'] == 'POST':
        '''
        Request body is being replaced. To do this, update the following
        three fields:
            1) body.action to 'replace'
            2) body.encoding to the encoding of the new data.

               Set to one of the following values:

                   text - denotes that the generated body is in text format.
                       Lambda@Edge will propagate this as is.
                   base64 - denotes that the generated body is base64 encoded.
                       Lambda@Edge will base64 decode the data before sending
                       it to the origin.
            3) body.data to the new body.
        '''
        request['body']['action'] = 'replace'
        request['body']['encoding'] = 'text'
        request['body']['data'] = getUpdatedBody(request)
    return request

def getUpdatedBody(request):
    # HTTP body is always passed as base64-encoded string. Decode it
    body = base64.b64decode(request['body']['data'])

    # HTML forms send data in query string format. Parse it
    params = {k: v[0] for k, v in parse_qs(body).items()}

    # For demonstration purposes, we're adding one more param

    # You can put your custom logic here. For example, you can truncate long
    # bodies from malicious requests
    params['new-param-name'] = 'new-param-value'
    return urlencode(params)
```

Lambda 함수에 대한 요구 사항 및 제한 사항

CloudFront에서 Lambda 함수를 사용할 경우 요구 사항 및 제한 사항은 다음 단원을 참조하십시오.

주제

- CloudFront 배포 및 연결 (p. 358)
- Lambda 함수에 대한 CloudFront 트리거 (p. 358)
- CloudWatch Logs (p. 358)
- 헤더 (p. 358)
- HTTP Status Codes (p. 360)
- Lambda 기능 지원 실행 시간 및 구성 (p. 360)
- 할당량 (p. 361)
- Microsoft Smooth Streaming (p. 362)
- 네트워크 액세스 (p. 362)
- 쿼리 문자열 매개 변수 (p. 362)
- 본문 포함 옵션이 적용된 본문에 대한 최대 크기 (p. 363)
- 태그 지정 (p. 363)
- URI (p. 363)
- URI 및 쿼리 문자열 인코딩 (p. 363)

CloudFront 배포 및 연결

- 또 다른 AWS 계정이 소유한 CloudFront 배포에는 Lambda 함수를 연결할 수 없습니다.

Lambda 함수에 대한 CloudFront 트리거

- 트리거는 `$LATEST` 또는 별칭이 아니라 번호가 지정된 버전에 대해서만 추가할 수 있습니다.
- 트리거는 미국 동부(버지니아 북부) 리전의 함수에 대해서만 추가할 수 있습니다.
- 트리거를 추가하려면 서비스 보안 주체 `lambda.amazonaws.com` 및 `edgelambda.amazonaws.com`에서 Lambda 함수와 연결된 IAM 실행 역할을 수임할 수 있어야 합니다. 자세한 내용은 Lambda@Edge에 대한 IAM 권한 및 역할 설정 (p. 292) 단원을 참조하십시오.

CloudWatch Logs

Amazon CloudWatch Logs 할당량(이전에는 제한이라고 함)에 대한 자세한 내용은 Amazon CloudWatch 사용 설명서의 CloudWatch Logs 할당량을 참조하십시오.

헤더

Lambda@Edge에 헤더를 사용할 경우 다음과 같은 요구 사항과 제한 사항에 주의하십시오.

주제

- 블랙리스트 헤더 (p. 358)
- 읽기 전용 헤더 (p. 359)
- CloudFront-* 헤더 (p. 359)

블랙리스트 헤더

블랙리스트에 등록된 헤더는 밖으로 표시되지 않으며 Lambda@Edge 함수가 추가할 수 없습니다. Lambda 함수가 블랙리스트에 등록된 헤더를 추가하는 경우 요청은 CloudFront 확인에 실패하고 CloudFront는 HTTP 상태 코드 502(잘못된 게이트웨이)를 최종 사용자에게 반환합니다.

- `Connection`
- `Expect`
- `Keep-Alive`
- `Proxy-Authenticate`
- `Proxy-Authorization`
- `Proxy-Connection`
- `Trailer`
- `Upgrade`
- `X-Accel-Buffering`
- `X-Accel-Charset`
- `X-Accel-Limit-Rate`
- `X-Accel-Redirect`
- `X-Amz-Cf-*`
- `X-Amzn-*`
- `X-Cache`

- `X-Edge-*`
- `X-Forwarded-Proto`
- `X-Real-IP`

읽기 전용 헤더

읽기 전용 헤더는 읽을 수는 있지만 편집할 수 없는 헤더입니다. 이러한 헤더는 CloudFront 캐싱 로직에 대한 입력으로 사용할 수 있으며, Lambda 함수가 헤더 값을 읽을 수 있지만 값을 변경할 수는 없습니다. Lambda 함수가 읽기 전용 헤더를 추가 또는 편집하는 경우 요청은 CloudFront 확인에 실패하고 CloudFront는 HTTP 상태 코드 502(잘못된 게이트웨이)를 최종 사용자에게 반환합니다.

CloudFront 최종 사용자 요청 이벤트의 읽기 전용 헤더

- `Content-Length`
- `Host`
- `Transfer-Encoding`
- `Via`

CloudFront 오리진 요청 이벤트의 읽기 전용 헤더

- `Accept-Encoding`
- `Content-Length`
- `If-Modified-Since`
- `If-None-Match`
- `If-Range`
- `If-Unmodified-Since`
- `Transfer-Encoding`
- `Via`

CloudFront 오리진 응답 이벤트의 읽기 전용 헤더

- `Transfer-Encoding`
- `Via`

CloudFront 최종 사용자 응답 이벤트의 읽기 전용 헤더

- `Content-Encoding`
- `Content-Length`
- `Transfer-Encoding`
- `Warning`
- `Via`

CloudFront-* 헤더

Lambda 함수는 다음 헤더를 읽거나, 편집, 제거 또는 추가할 수 있습니다.

- `CloudFront-Forwarded-Proto`
- `CloudFront-Is-Desktop-Viewer`

- `CloudFront-Is-Mobile-Viewer`
- `CloudFront-Is-SmartTV-Viewer`
- `CloudFront-Is-Tablet-Viewer`
- `CloudFront-Viewer-Country`

다음을 참조하십시오.

- CloudFront에서 이러한 헤더를 추가하려는 경우 이러한 헤더를 기반으로 캐시하도록 CloudFront를 구성해야 합니다. 지정된 헤더를 기반으로 캐시하도록 CloudFront를 구성하는 방법에 대한 자세한 내용은 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 주제의 선택한 요청 헤더 기반의 캐시 (p. 46) 단원을 참조하십시오.
- CloudFront는 최종 사용자 요청 이벤트 뒤에 헤더를 추가합니다.
- 최종 사용자가 이러한 이름이 있는 헤더를 추가하면 CloudFront는 헤더 값을 덮어씁니다.
- 최종 사용자 이벤트에 대해 `CloudFront-Viewer-Country`가 블랙리스트에 등록되었습니다. 블랙리스트에 등록된 헤더는 밖으로 표시되지 않으며 Lambda@Edge 함수가 추가할 수 없습니다. Lambda 함수가 블랙리스트에 등록된 헤더를 추가하는 경우 요청은 CloudFront 확인에 실패하고 CloudFront는 HTTP 상태 코드 502(잘못된 게이트웨이)를 최종 사용자에게 반환합니다.

자세한 내용은 다음 예를 참조하십시오.

- 예: 최종 사용자 요청을 국가별 URL로 리디렉션 (p. 342)
- 예: 디바이스를 기반으로 객체의 다양한 버전 서비스 (p. 343)

HTTP Status Codes

오리진이 400 이상의 HTTP 상태 코드를 반환하는 경우 CloudFront는 최종 사용자 응답 이벤트에 대해 Lambda 함수를 실행하지 않습니다. 또한 요청이 성공하면 뷰어 응답 이벤트에서 HTTP 상태 코드를 수정할 수 없습니다.

그러나 HTTP 상태 코드 4xx와 5xx를 포함한 오리진 응답 오류의 경우 Lambda 함수를 실행할 수 있습니다. 자세한 내용은 오리진 응답 트리거에서 HTTP 응답 업데이트 (p. 329) 단원을 참조하십시오.

Lambda 기능 지원 실행 시간 및 구성

- Lambda@Edge는 다음과 같은 실행 시간으로 Lambda 함수를 지원합니다.
  - Python 3.8
  - Python 3.7
  - Node.js 12
  - Node.js 10
  - Node.js 8 및 Node.js 6

    Note

    Node.js 버전 8 및 버전 6은 수명 주기가 다 되었습니다. 이러한 실행 시간으로 함수를 만들거나 업데이트할 수 없습니다. 이러한 실행 시간 중 하나가 있는 기존 함수가 있는 경우에도 이러한 함수를 CloudFront 배포와 연결할 수 있으며 배포와 이미 연결된 함수도 계속 실행됩니다. 하지만 함수를 최신 버전의 Node.js로 이동하는 것이 좋습니다. 자세한 내용은 AWS Lambda Developer Guide의 실행 시간 지원 정책과 GitHub의 Node.js 릴리스 일정을 참조하십시오.
- Lambda 함수가 VPC 내의 리소스에 액세스하도록 구성할 수는 없습니다.
- 또 다른 AWS 계정이 소유한 CloudFront 배포에는 Lambda 함수를 연결할 수 없습니다.
- AWS Lambda 함수 배달 못한 편지 대기열은 지원되지 않습니다.

- AWS Lambda 환경 변수는 지원되지 않습니다.
- AWS Lambda 계층이 있는 함수는 지원되지 않습니다.
- AWS X-Ray 사용은 지원되지 않습니다.
- AWS Lambda 예약된 동시성 및 프로비저닝된 동시성은 지원되지 않습니다.

할당량

이 섹션의 할당량은 Lambda@Edge에 적용됩니다. 이러한 할당량은 기본 CloudFront 및 Lambda 할당량에 추가하여 적용됩니다. 기본 할당량은 이 가이드의 할당량 (p. 447) 및 AWS Lambda Developer Guide의 할당량을 참조하십시오.

> **Note**
>
> Lambda는 계정 할당량 내에서 트래픽 증가에 대응하여 용량을 동적으로 확장합니다. 자세한 내용은 AWS Lambda Developer Guide의 함수 규모 조정을 참조하십시오.

추가로 Lambda@Edge 함수 사용 시 기타 제한 사항이 있습니다. 자세한 내용은 Lambda 함수에 대한 요구 사항 및 제한 사항 (p. 357) 단원을 참조하십시오.

이벤트 유형에 따라 다른 할당량

엔터티	오리진 요청 및 응답 이벤트 할당량	최종 사용자 요청 및 응답 이벤트 할당량
함수 메모리 크기	Lambda 할당량과 동일	128MB
함수 제한 시간. 이 함수는 AWS 리전에서 Amazon S3 버킷, DynamoDB 테이블 또는 Amazon EC2 인스턴스와 같은 리소스에 대한 네트워크 호출을 수행할 수 있습니다.	30초	5초
헤더 및 본문을 포함하여 Lambda 함수에서 생성되는 응답의 크기입니다.	1MB	40KB
Lambda 함수 및 포함된 라이브러리의 최대 압축 크기	50MB	1MB

기타 할당량

엔터티	기본 할당량
트리거를 생성할 수 있는 AWS 계정당 배포 수입니다.	25 더 높은 할당량 요청
배포당 트리거 수	100 더 높은 할당량 요청
초당 요청	10,000(각 리전에서) 더 높은 할당량 요청
동시 실행 자세한 내용은 AWS Lambda Developer Guide의 함수 규모 조정을 참조하십시오.	1,000(각 리전에서) 더 높은 할당량 요청

URI 및 쿼리 문자열의 크기 할당량

Lambda@Edge 함수에서 URI 또는 쿼리 문자열에 액세스하거나 업데이트할 때 URI의 총 길이는 쿼리 문자열을 포함하여 8,192자 미만이어야 합니다.

본문 포함 옵션이 적용된 요청 본문에 대한 크기 할당량

Include Body(본문 포함) 옵션을 선택해 Lambda@Edge 함수에 요청 본문을 노출하면 노출 또는 교체된 본문 부분에 다음 크기 할당량이 적용됩니다.

> **Note**
>
> 본문은 노출 전 항상 Lambda@Edge로 base64 인코딩됩니다.

본문을 Lambda 함수에 노출 시 크기 할당량

요청 본문이 크면 노출되기 전에 Lambda@Edge가 다음과 같이 자릅니다.

- 최종 사용자 요청의 경우 본문은 40KB에서 잘립니다.
- 오리진 요청의 경우 본문은 1MB에서 잘립니다.

Lambda 함수에서 요청 본문 반환 시 크기 할당량

요청 본문에 읽기 전용으로 액세스하는 경우 전체 원본 요청 본문이 오리진으로 반환됩니다. 그러나 요청 본문을 바꾸도록 선택한 경우 Lambda 함수에서 반환 시 다음 본문 크기 할당량이 적용됩니다.

본문 인코딩 유형	허용되는 본문 크기: 최종 사용자 요청	허용되는 본문 크기: 오리진 요청
text	40KB	1MB
base64	53.2KB	1.33MB

Microsoft Smooth Streaming

Microsoft Smooth Streaming 형식으로 트랜스코딩한 미디어 파일의 온디맨드 비디오(VOD) 스트리밍에 사용하는 CloudFront 배포에 대해서는 트리거를 생성할 수 없습니다.

네트워크 액세스

오리진 요청 및 응답 이벤트는 물론 최종 사용자 요청 및 응답 이벤트가 트리거한 함수는 인터넷 상 리소스 및 AWS 서비스(예: Amazon S3 버킷, DynamoDB 테이블, Amazon EC2 인스턴스)에 대해 네트워크 호출을 생성할 수 있습니다.

쿼리 문자열 매개 변수

- Lambda 함수의 쿼리 문자열에 액세스하려면 `event.Records[0].cf.request.querystring`을 사용합니다.
- 함수는 최종 사용자 및 오리진 요청 이벤트에 대한 쿼리 문자열을 업데이트할 수 있습니다. 업데이트된 쿼리 문자열에는 공백, 제어 문자 또는 조각 ID(#)가 포함될 수 없습니다.
- 함수는 오리진 및 최종 사용자 응답 이벤트에 대한 쿼리 문자열만 읽을 수 있습니다.
- 쿼리 문자열 파라미터를 기반으로 캐시하도록 CloudFront를 구성하면 함수가 쿼리 문자열에 액세스할 수 있는지 여부에 영향을 미칩니다.

  - 최종 사용자 요청 및 응답 이벤트 – Query String Forwarding and Caching (p. 49)에 대한 설정과 상관없이 함수가 쿼리 문자열에 액세스할 수 있습니다.
  - 오리진 요청 및 응답 이벤트 – Query String Forwarding and Caching (p. 49)이 Forward All, Cache Based on Whitelist(모두 전달, 화이트리스트를 기반으로 캐싱) 또는 Forward All, Cache Based on All(모두 전달, 모두를 기반으로 캐싱)로 설정된 경우에만 함수가 쿼리 문자열에 액세스할 수 있습니다.
- URI 및 쿼리 문자열에 퍼센트 인코딩을 사용하는 것이 좋습니다. 자세한 내용은 URI 및 쿼리 문자열 인코딩 (p. 363) 단원을 참조하십시오.
- URI(`event.Records[0].cf.request.uri`) 및 쿼리 문자열 (`event.Records[0].cf.request.querystring`)의 총 크기는 8,192자 미만이어야 합니다.

자세한 내용은 쿼리 문자열 파라미터 기반의 콘텐츠 캐싱 (p. 196) 단원을 참조하십시오.

본문 포함 옵션이 적용된 본문에 대한 최대 크기

크기 할당량을 포함하여 모든 Lambda@Edge 할당량(이전에는 제한이라고 함)에 대한 자세한 내용은 할당량 (p. 361) 단원을 참조하십시오.

태그 지정

Amazon CloudFront 및 AWS Lambda를 포함한 일부 AWS 서비스는 서비스 내의 리소스에 태그 추가를 지원합니다. 하지만 지금은 Lambda@Edge 리소스에 태그를 적용할 수 없습니다. CloudFront의 태깅에 대해 자세히 알아보려면 Amazon CloudFront 배포에 태그 지정 (p. 424) 단원을 참조하십시오.

URI

함수가 요청의 URI를 변경하는 경우 이로 인해 요청 또는 요청이 전달되는 오리진에 대한 캐시 동작이 변경되지 않습니다.

URI 및 쿼리 문자열 인코딩

Lambda 함수에서는 URI 및 쿼리 문자열을 UTF-8으로 인코딩해야 합니다. (퍼센트 인코딩은 UTF-8 인코딩과 호환됩니다.) CloudFront 및 Lambda 의 동작은 다음에 따라 결정됩니다.

- CloudFront가 최종 사용자의 요청에서 수신하는 URI 및 쿼리 문자열의 인코딩
- 최종 사용자 요청 또는 오리진 요청 이벤트에서 트리거되는 함수에 따라 URI 또는 쿼리 문자열이 변경되는지 여부

값이 UTF-8으로 인코딩됨

CloudFront는 값을 변경하지 않고 Lambda 함수에 전달합니다.

값이 ISO 8859-1으로 인코딩됨

CloudFront는 ISO 8859-1 문자 인코딩을 UTF-8 인코딩으로 변환한 후 값을 Lambda 함수에 전달합니다.

다른 특정 문자 인코딩을 사용하여 값이 인코딩됨

다른 문자 인코딩을 사용하여 값이 인코딩되는 경우 CloudFront는 값이 ISO 8859-1으로 인코딩된 것으로 가정하고 ISO 8859-1 인코딩에서 UTF-8 인코딩으로 변환하려고 시도합니다.

> **Important**
>
> 변환된 버전은 원래 요청에 있는 값의 부정확한 해석일 수 있습니다. 이 경우 Lambda 함수 또는 오리진에서 의도하지 않은 결과가 생성될 수 있습니다.

CloudFront가 오리진 서버에 전달하는 값은 최종 사용자 요청 또는 오리진 요청 이벤트에서 트리거된 함수가 URI 또는 쿼리 문자열을 변경하는지 여부에 따라 결정됩니다.

- 함수가 URI 또는 쿼리 문자열을 변경하지 않는 경우 – CloudFront는 CloudFront가 최종 사용자의 요청에서 수신한 값을 오리진 서버에 전달합니다.
- 함수가 URI 또는 쿼리 문자열을 변경하는 경우 – CloudFront는 UTF-8으로 인코딩된 값을 전달합니다.

두 경우 모두 동작은 최종 사용자 요청의 문자 인코딩에 따라 영향을 받습니다.

보고서, 액세스 로그, 모니터링

이 단원에는 CloudFront에 대한 모니터링 및 보고서 옵션에 대한 세부 정보를 제공하는 주제가 포함되어 있습니다. CloudFront 배포에 대한 사용량과 활동을 볼 수 있는 다양한 보고서가 제공됩니다. 여기에는 결제 보고서, 캐시 통계, 인기 있는 콘텐츠, 상위 참조자 등이 있습니다. 또한 CloudFront 콘솔에서 직접 또는 CloudTrail 및 CloudWatch와 같은 도구를 사용하여 Lambda@Edge 활동을 비롯한 CloudFront를 모니터링하고 추적할 수 있습니다.

주제

- CloudFront에 대한 AWS 결제 및 사용 보고서 (p. 365)
- 콘솔의 CloudFront 보고서 (p. 369)
- AWS Config를 통해 구성 변경 추적 (p. 392)
- 액세스 로그 구성 및 사용 (p. 393)
- AWS CloudTrail를 사용하여 CloudFront API로 전송된 요청 캡처 (p. 410)
- Amazon CloudWatch를 사용하여 CloudFront 모니터링 (p. 416)
- Amazon CloudFront 배포에 태그 지정 (p. 424)

CloudFront에 대한 AWS 결제 및 사용 보고서

AWS에서는 CloudFront에 대해 다음 두 가지 사용 보고서를 제공합니다.

- 결제 보고서는 CloudFront를 포함해 사용 중인 AWS 서비스에 대한 모든 활동을 세부적으로 보여 줍니다. 자세한 내용은 CloudFront에 대한 AWS 결제 보고서 (p. 365) 단원을 참조하십시오.
- 사용 보고서는 특정 서비스에 대한 활동을 시간, 일 또는 월 기준으로 요약합니다. 또한 CloudFront 사용을 그래픽으로 표시하는 사용 차트가 포함되어 있습니다. 자세한 내용은 CloudFront에 대한 AWS 사용 보고서 (p. 366) 단원을 참조하십시오.

이러한 보고서를 이해하는 데 도움이 되도록 자세한 내용은 CloudFront에 대한 AWS 청구서 및 AWS 사용 보고서 해석 (p. 367) 단원을 참조하십시오.

> **Note**
>
> 기타 AWS 서비스와 마찬가지로 CloudFront는 사용한 만큼만 과금합니다. 자세한 내용은 CloudFront 요금 (p. 8) 단원을 참조하십시오.

주제

- CloudFront에 대한 AWS 결제 보고서 (p. 365)
- CloudFront에 대한 AWS 사용 보고서 (p. 366)
- CloudFront에 대한 AWS 청구서 및 AWS 사용 보고서 해석 (p. 367)

CloudFront에 대한 AWS 결제 보고서

AWS Management 콘솔의 청구서 페이지에서 서비스별로 나열된 AWS 사용 및 요금을 볼 수 있습니다.

CSV 형식으로 보고서의 보다 세부적인 버전을 다운로드할 수도 있습니다. 세부 결제 보고서에는 CloudFront에 적용되는 다음 값이 포함됩니다.

- ProductCode — `AmazonCloudFront`
- UsageType - 다음 값 중 하나입니다.
  - 데이터 전송의 유형을 식별하는 코드입니다.
  - `Invalidations`
  - `SSL-Cert-Custom`

  자세한 내용은 CloudFront에 대한 AWS 청구서 및 AWS 사용 보고서 해석 (p. 367) 단원을 참조하십시오.
- ItemDescription - UsageType의 청구 요금에 대한 설명입니다.
- Usage Start Date/Usage End Date(사용 시작 날짜/사용 종료 날짜) - 사용이 적용되는 날짜로, UTC(협정 세계시)를 따릅니다.
- 사용량 - 다음 값 중 하나입니다.
  - 지정된 기간 동안 요청 수
  - 전송된 데이터의 양(기가바이트 단위)
  - 무효화된 객체 수
  - 활성화된 CloudFront 배포와 SSL 인증서가 연결된 개월 수를 계산한 합계입니다. 예를 들어 한 달 내내 활성화된 배포와 연결된 인증서가 하나 있고, 15일 동안 활성화된 배포와 연결된 또 다른 인증서가 있는 경우 이 값은 1.5가 됩니다.

요약 결제 정보를 표시하고 세부 결제 보고서를 다운로드하는 방법

1. https://console.aws.amazon.com/console/home에서 AWS Management 콘솔에 로그인합니다.
2. 제목 표시줄에서 IAM 사용자 이름을 클릭하고 대금 및 비용 관리를 클릭합니다.
3. 탐색 창에서 청구서를 클릭합니다.
4. CloudFront에 대한 요약 정보를 보려면 세부 정보 아래에서 CloudFront를 클릭합니다.
5. CSV 형식으로 세부 결제 보고서를 다운로드하려면 CSV 다운로드를 클릭하고 화면에 나타나는 메시지에 따라 보고서를 저장합니다.

CloudFront에 대한 AWS 사용 보고서

AWS는 결제 보고서보다 세부적이지만 CloudFront 액세스 로그보다 간략한 CloudFront 사용 보고서를 제공합니다. 사용 보고서는 시간, 일 또는 월 기준으로 누적 사용량 데이터를 제공하며, 오스트레일리아 지역 외부로 전송된 데이터 같은 지역 및 사용 유형에 따른 작업을 나열합니다.

CloudFront 사용 보고서에는 다음 값이 포함됩니다.

- 서비스 - `AmazonCloudFront`
- 작업 - HTTP 메서드입니다. 값에는 `DELETE`, `GET`, `HEAD`, `OPTIONS`, `PATCH`, `POST` 및 `PUT`가 포함됩니다.
- UsageType - 다음 값 중 하나입니다.
  - 데이터 전송의 유형을 식별하는 코드입니다.
  - `Invalidations`
  - `SSL-Cert-Custom`

  자세한 내용은 CloudFront에 대한 AWS 청구서 및 AWS 사용 보고서 해석 (p. 367) 단원을 참조하십시오.
- 리소스 - 사용과 연결된 CloudFront 배포의 ID 또는 CloudFront 배포와 연결한 SSL 인증서의 인증서 ID입니다.

- StartTime/EndTime - 사용이 적용되는 날짜로, UTC(협정 세계시)를 따릅니다.
- UsageValue - (1) 지정된 기간 동안 요청 수 또는 (2) 바이트 단위로 전송된 데이터의 양입니다.

Amazon S3을 CloudFront에 대한 오리진으로 사용하는 경우 Amazon S3에 대해서도 사용 보고서를 실행해 보십시오. 그러나 Amazon S3을 CloudFront 배포를 위한 오리진이 아닌 다른 목적으로 사용하는 경우 CloudFront 사용에 어떤 부분이 적용되는지 명확하지 않을 수 있습니다.

> Tip
>
> CloudFront가 객체에 대해 수신하는 모든 요청에 대한 자세한 내용을 보려면 배포에 대한 CloudFront 액세스 로그를 활성화하십시오. 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

CloudFront 또는 Amazon S3에 대한 사용 보고서를 다운로드하는 방법

1. https://console.aws.amazon.com/console/home에서 AWS Management 콘솔에 로그인합니다.
2. 제목 표시줄에서 IAM 사용자 이름을 클릭하고 대금 및 비용 관리를 클릭합니다.
3. 탐색 창에서 보고서를 클릭합니다.
4. AWS 사용 보고서 아래에서 AWS 사용 보고서를 클릭합니다.
5. 서비스 목록에서 CloudFront 또는 Amazon Simple Storage Service를 클릭합니다.
6. 사용할 설정을 선택합니다.
   - 사용 유형 — CloudFront 사용 유형에 대한 자세한 설명은 the section called “CloudFront에 대한 AWS 청구서 및 AWS 사용 보고서 해석” (p. 367)을 참조하십시오.

     Amazon S3의 경우 모든 사용 유형을 선택합니다.
   - 작업 - 모든 작업을 선택합니다.
   - 기간 - 보고서에 포함할 기간을 선택합니다.
   - 보고서 세부 수준 - 보고서에 시간, 일 또는 월 기준으로 소계를 포함할지 여부를 선택합니다.
7. 원하는 형식에 해당하는 다운로드 버튼을 클릭합니다.
8. 화면에 나타나는 메시지에 따라 보고서를 보거나 저장합니다.

CloudFront에 대한 AWS 청구서 및 AWS 사용 보고서 해석

CloudFront 서비스에 대한 AWS 청구서에는 즉시 눈에 띄지 않을 수 있는 코드 및 약어가 포함됩니다. 다음 표의 첫째 열에는 청구서에 나타나는 항목이 나열되고 각 항목의 의미가 설명되어 있습니다.

또한 CloudFront에 대한 AWS 청구서보다 자세한 정보가 포함되어 있는 CloudFront에 대한 AWS 사용 보고서를 가져올 수 있습니다. 표의 둘째 열에는 사용 보고서에 나타나는 항목이 나열되고, 청구서 항목과 사용 보고서 항목 간의 상관 관계가 표시됩니다.

두 열 모두에 있는 대부분의 코드에는 활동의 위치를 나타내는 두 글자 약어가 포함됩니다. 다음 표에서 코드의 `region`은 AWS 청구서와 사용 보고서에 있는 다음 두 글자 약어 중 하나로 바뀝니다.

- AP: 홍콩, 필리핀, 한국, 대만 및 싱가포르(아시아 태평양)
- AU: 오스트레일리아
- CA: 캐나다
- EU: 유럽 및 이스라엘
- IN: 인도
- JP: 일본
- ME: 중동

- SA: 남미
- US: 미국
- ZA: 남아프리카

지역별 요금에 대한 자세한 내용은 Amazon CloudFront 요금을 참조하십시오.

> **Note**
>
> 이 표에는 Amazon S3 버킷에서 CloudFront 엣지로 객체를 전송하는 데 드는 요금이 포함되어 있지 않습니다. 이 요금은 존재하는 경우 AWS 청구서의 AWS Data Transfer(AWS 데이터 전송) 부분에 나타납니다.

CloudFront 청구서의 항목	CloudFront 사용 보고서의 사용 유형 열에 있는 값
region-DataTransfer-Out-Bytes CloudFront가 웹 및 RTMP 배포를 위해 서비스하는 바이트의 합계: • 웹 배포: 사용자 `GET` 및 `HEAD` 요청에 대한 응답으로 *region*의 CloudFront 엣지에서 서비스되는 총 바이트 • RTMP 배포: *region*의 CloudFront 엣지에서 최종 사용자에게 전송되는 총 바이트	웹 배포: • *region*-Out-Bytes-HTTP-Static: TTL이 3600초보다 크거나 같은 객체에 대해 HTTP를 통해 서비스되는 바이트 • *region*-Out-Bytes-HTTPS-Static: TTL이 3600초보다 크거나 같은 객체에 대해 HTTPS를 통해 서비스되는 바이트 • *region*-Out-Bytes-HTTP-Dynamic: TTL이 3600초보다 작은 객체에 대해 HTTP를 통해 서비스되는 바이트 • *region*-Out-Bytes-HTTPS-Dynamic: TTL이 3600초보다 작은 객체에 대해 HTTPS를 통해 서비스되는 바이트 • *region*-Out-Bytes-HTTP-Proxy: `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대한 응답으로 HTTP를 통해 CloudFront에서 최종 사용자에게 반환되는 바이트 • *region*-Out-Bytes-HTTPS-Proxy: `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대한 응답으로 HTTPS를 통해 CloudFront에서 최종 사용자에게 반환되는 바이트 RTMP 배포: • *region*-FMS-Out-Bytes
region-DataTransfer-Out-OBytes 웹 배포만: `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대한 응답으로 CloudFront 엣지에서 오리진으로 전송되는 총 바이트 요금에는 WebSocket 데이터의 클라이언트에서 서버로의 전송에 따른 금액이 포함됩니다.	*region*-Out-OBytes-HTTP-Proxy `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대한 응답으로 HTTP를 통해 CloudFront 엣지에서 오리진으로 전송되는 총 바이트 *region*-Out-OBytes-HTTPS-Proxy `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대한 응답으로 HTTPS를 통해 CloudFront 엣지에서 오리진으로 전송되는 총 바이트
region-Requests-Tier1 웹 배포만: HTTP `GET` 및 `HEAD` 요청 수	*region*-Requests-HTTP-Static TTL이 3600초보다 작은 객체에 대해 서비스되는 HTTP `GET` 및 `HEAD` 요청의 수 *region*-Requests-HTTP-Dynamic TTL이 3600초보다 작은 객체에 대해 서비스되는 HTTP `GET` 및 `HEAD` 요청의 수

CloudFront 청구서의 항목	CloudFront 사용 보고서의 사용 유형 열에 있는 값
region-Requests-Tier2-HTTPS 웹 배포만: HTTPS `GET` 및 `HEAD` 요청 수	*region*-Requests-HTTPS-Static TTL이 3600초보다 작은 객체에 대해 서비스되는 HTTPS `GET` 및 `HEAD` 요청의 수 *region*-Requests-HTTPS-Dynamic TTL이 3600초보다 작은 객체에 대해 서비스되는 HTTPS `GET` 및 `HEAD` 요청의 수
region-Requests-HTTP-Proxy 웹 배포만: CloudFront가 오리진에 전달하는 HTTP `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청의 수	*region*-Requests-HTTP-Proxy CloudFront 청구서의 해당 항목과 같음
region-Requests-HTTPS-Proxy 웹 배포만: CloudFront가 오리진에 전달하는 HTTPS `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청의 수	*region*-Requests-HTTPS-Proxy CloudFront 청구서의 해당 항목과 같음
region-Requests-HTTPS-Proxy-FLE 웹 배포만: CloudFront가 필드 레벨 암호화를 통해 처리된 오리진으로 전송한 HTTPS `DELETE`, `OPTIONS`, `PATCH` 및 `POST` 요청 수.	*region*-Requests-HTTPS-Proxy-FLE CloudFront 청구서의 해당 항목과 같음
무효화 웹 배포만: 객체를 무효화(CloudFront 엣지에서 객체 제거)하는 데 드는 요금. 자세한 내용은 파일 무효화에 대한 요금 결제 (p. 86) 단원을 참조하십시오.	무효화 CloudFront 청구서의 해당 항목과 같음
SSL-Cert-Custom 웹 배포만: SSL 인증서를 기본 CloudFront SSL 인증서 대신 example.com 같은 CloudFront 대체 도메인 이름 및 CloudFront에서 배포에 할당한 도메인 이름과 함께 사용하는 데 드는 요금	SSL-Cert-Custom CloudFront 청구서의 해당 항목과 같음

콘솔의 CloudFront 보고서

CloudFront 콘솔에는 다음을 비롯한 CloudFront 활동에 대한 다양한 보고서가 있습니다.

- CloudFront 캐시 통계 보고서 (p. 370)
- CloudFront 인기 객체 보고서 (p. 370)
- CloudFront 상위 참조자 보고서 (p. 370)
- CloudFront 사용 보고서 (p. 370)
- CloudFront 최종 사용자 보고서 (p. 370)

이러한 보고서 중 대부분은 CloudFront 액세스 로그의 데이터에 기초합니다. 이 액세스 로그에는 CloudFront 에서 수신하는 모든 사용자 요청에 대한 세부 정보가 포함되어 있습니다. 보고서를 보기 위해 액세스 로그를 활성화할 필요가 없습니다. 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오. CloudFront 사용 보고서는 CloudFront용 AWS 사용 보고서에 기초하므로 별도의 구성이 필요하지 않습니다. 자세한 내용은 CloudFront에 대한 AWS 사용 보고서 (p. 366) 단원을 참조하십시오.

CloudFront 캐시 통계 보고서

CloudFront 캐시 통계 보고서에는 다음 정보가 포함됩니다.

- 전체 요청 – 해당 기간 동안의 전체 HTTP 상태 코드(예: 200 또는 404)와 전체 메서드(예: GET, HEAD, POST)의 총 요청 수입니다.
- 최종 사용자 요청에 대한 결과 유형별 비율 – 선택한 CloudFront 배포에 대한 총 최종 사용자 요청 수에서 적중, 누락 및 오류 수를 백분율로 표시합니다.
- 최종 사용자에게 전송된 바이트 수 – 총 바이트 수와 누락 바이트 수를 표시합니다.
- HTTP 상태 코드 – HTTP 상태 코드에 의한 최종 사용자 요청을 표시합니다.
- 다운로드가 완료되지 않은 GET 요청 비율– 요청한 객체의 다운로드가 완료되지 않은 최종 사용자 GET 요청을 총 요청의 백분율로 표시합니다.

자세한 내용은 CloudFront 캐시 통계 보고서 (p. 371) 단원을 참조하십시오.

CloudFront 인기 객체 보고서

CloudFront 인기 객체 보고서에는 객체 요청 수, 적중 및 누락 수, 적중률, 누락 바이트 수, 제공된 총 바이트 수, 완료되지 않은 다운로드 수, HTTP 상태 코드별 요청 수(2xx, 3xx, 4xx, and 5xx) 등 객체에 대한 통계와 상위 50개 인기 객체가 나열됩니다.

자세한 내용은 CloudFront 인기 객체 보고서 (p. 374) 단원을 참조하십시오.

CloudFront 상위 참조자 보고서

CloudFront 상위 참조자 보고서에는 상위 25개 참조자, 참조자의 요청 수, 그리고 지정된 기간 동안 총 요청 수의 백분율로 나타낸 참조자의 요청 수가 포함됩니다.

자세한 내용은 CloudFront 상위 참조자 보고서 (p. 378) 단원을 참조하십시오.

CloudFront 사용 보고서

CloudFront 사용 보고서에는 다음 정보가 포함됩니다.

- 요청 수 – 지정된 CloudFront 배포에 대한 매 시간 간격 동안 선택한 리전의 엣지 로케이션에서 CloudFront 이 응답하는 요청 수를 표시합니다.
- 프로토콜에서 전송한 데이터 및 대상에서 전송한 데이터 – 둘 다 지정된 CloudFront 배포에 대한 매 시간 간격 동안 선택한 리전의 CloudFront 엣지 위치에서 전송한 총 데이터 양을 표시합니다. 다음과 같이 데이터를 구분합니다.
  - 프로토콜별 – HTTP 또는 HTTPS 프로토콜로 데이터를 구분합니다
  - 대상별 – 대상(사용자 또는 오리진)으로 데이터를 구분합니다

자세한 내용은 CloudFront 사용 보고서 (p. 380) 단원을 참조하십시오.

CloudFront 최종 사용자 보고서

CloudFront 최종 사용자 보고서에는 다음 정보가 포함됩니다.

- 디바이스 – 사용자가 콘텐츠에 액세스하는 데 사용하는 디바이스 유형(예: 데스크톱 또는 모바일)을 표시합니다.

- 브라우저 – 사용자가 콘텐츠에 액세스할 때 자주 사용하는 브라우저의 이름이나 버전(예: Chrome 또는 Firefox)을 표시합니다.
- 운영 체제 – 최종 사용자가 콘텐츠에 액세스할 때 자주 실행하는 운영 체제의 이름이나 버전(예: Linux, Mac OS X 또는 Windows)을 표시합니다.
- 위치 – 콘텐츠에 자주 액세스하는 최종 사용자의 국가별 또는 미국 주/지역별 위치를 표시합니다.

자세한 내용은 CloudFront Viewers Reports (p. 385) 단원을 참조하십시오.

CloudFront 캐시 통계 보고서

Amazon CloudFront 콘솔을 사용하여 CloudFront 엣지 로케이션과 관련된 통계 그래프를 표시할 수 있습니다. 이러한 통계에 대한 데이터는 CloudFront 액세스 로그와 동일한 소스에서 가져옵니다. 한 시간 또는 하루 기준의 데이터 지점을 포함해 지난 60일 동안 지정된 날짜 범위에 대한 차트를 표시할 수 있습니다. 일반적으로 한 시간 전까지 CloudFront에서 수신한 요청에 대한 데이터를 볼 수 있지만, 이따금씩 데이터가 24시간 가량 지연될 수 있습니다.

> **Note**
>
> 캐시 통계를 보기 위해 액세스 로깅을 활성화할 필요가 없습니다.

CloudFront 캐시 통계를 표시하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 탐색 창에서 Cache Statistics(캐시 통계)를 클릭합니다.
3. CloudFront 캐시 통계 보고서 창의 시작일 및 종료일 필드에서 캐시 통계 차트를 표시할 날짜 범위를 선택합니다. 사용 가능한 범위는 세부 수준에 선택한 값에 따라 다릅니다.
   - 매일 – 하루에 데이터 지점 하나씩 포함하는 차트를 표시하려면 이전 60일에 해당하는 날짜 범위를 선택합니다.
   - 매 시간 – 한 시간에 데이터 지점 하나씩 포함하는 차트를 표시하려면 이전 60일 내에서 날짜 범위를 최대 14일까지 선택합니다.

   날짜 및 시간이 협정 세계시(UTC)로 표시됩니다.
4. 세부 수준의 경우 차트에 하루에 데이터 지점 하나씩 표시할지, 한 시간에 데이터 지점 하나씩 표시할지 지정합니다. 날짜 범위를 14일보다 길게 지정할 경우 한 시간에 데이터 지점 하나를 지정하는 옵션을 사용할 수 없습니다.
5. 최종 사용자에서는 최종 사용자 요청이 시작된 대륙을 선택하거나 All Locations(모든 위치)를 선택합니다. 캐시 통계 보고서에는 CloudFront에서 지정된 위치로부터 수신된 요청에 대한 데이터가 포함됩니다.
6. 배포 목록에서 사용 차트에 데이터를 표시할 배포를 선택합니다.
   - 개별 웹 배포 – 이 차트에는 선택한 CloudFront 웹 배포에 대한 데이터가 표시됩니다. 배포 목록은 배포에 대한 ID와 대체 도메인 이름(CNAME)이 있는 경우 이를 표시합니다. 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.
   - All Web Distributions(모든 웹 배포) – 이 차트에는 사용자가 삭제한 웹 배포를 제외한 현재 AWS 계정과 연결된 모든 웹 배포에 대해 합산된 데이터가 표시됩니다.
7. 업데이트를 클릭합니다.
8. 차트 안에서 일별 또는 시간별 데이터를 보려면 마우스 포인터를 데이터 지점 위로 이동합니다.
9. 전송된 데이터를 보여 주는 차트의 경우 각 차트에 대해 수직 눈금을 기가바이트, 메가바이트 또는 킬로바이트로 변경할 수 있습니다.

주제

- 데이터를 CSV 형식으로 다운로드 (p. 372)
- 캐시 통계 차트와 CloudFront 액세스 로그 데이터 간의 연결 (p. 373)

데이터를 CSV 형식으로 다운로드

캐시 통계 보고서를 CSV 형식으로 다운로드할 수 있습니다. 이 단원에서는 보고서를 다운로드하는 방법과 보고서의 값을 설명합니다.

캐시 통계 보고서를 CSV 형식으로 다운로드하려면

1. 캐시 통계 보고서를 표시한 상태에서 CSV를 클릭합니다.
2. 파일 이름 열기 대화 상자에서 파일을 열지 또는 저장할지 여부를 선택합니다.

보고서에 대한 정보

보고서의 처음 몇 줄에는 다음 정보가 포함됩니다.

버전

이 CSV 파일 형식의 버전입니다.

보고서

보고서의 이름입니다.

DistributionID

보고서가 실행된 배포의 ID 또는, 모든 배포에 대한 보고서를 실행한 경우 `ALL`입니다.

StartDateUTC

보고서 실행 기간에 해당하는 날짜 범위의 시작 일시를 협정 세계시(UTC)로 표시합니다.

EndDateUTC

보고서 실행 기간에 해당하는 날짜 범위의 종료 일시를 협정 세계시(UTC)로 표시합니다.

GeneratedTimeUTC

보고서를 실행 날짜 및 시간을 협정 세계시(UTC)로 표시합니다.

Granularity

보고서의 각 줄이 한 시간을 나타내는지 하루를 나타내는지 지정합니다.

ViewerLocation

최종 사용자 요청이 시작된 대륙, 도는 모든 위치에 대한 보고서를 다운로드하도록 선택한 경우 `ALL`입니다.

캐시 통계 보고서의 데이터

이 보고서에는 다음 값이 포함됩니다.

DistributionID

보고서가 실행된 배포의 ID 또는, 모든 배포에 대한 보고서를 실행한 경우 `ALL`입니다.

FriendlyName

배포의 대체 도메인 이름(CNAME)입니다(있는 경우). 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.

ViewerLocation

최종 사용자 요청이 시작된 대륙, 도는 모든 위치에 대한 보고서를 다운로드하도록 선택한 경우 `ALL`입니다.

TimeBucket

데이터가 적용되는 시간 또는 날짜를 협정 세계시(UTC)로 표시합니다.

RequestCount

해당 기간 동안의 전체 HTTP 상태 코드(예: 200 또는 404)와 전체 메서드(예: GET, HEAD, POST)의 총 요청 수입니다.

HitCount

CloudFront 엣지 캐시에서 객체가 제공되는 최종 사용자 요청 수입니다.

MissCount

현재 엣지 캐시에 객체가 없어서 CloudFront가 오리진에서 객체를 가져와야 하는 최종 사용자 요청 수입니다.

ErrorCount

오류가 발생하여 CloudFront에서 객체를 제공하지 못한 최종 사용자 요청 수입니다.

IncompleteDownloadCount

최종 사용자가 객체 다운로드를 시작했지만 완료하지 못한 최종 사용자 요청 수입니다.

HTTP2xx

HTTP 상태 코드가 2xx 값(성공)인 최종 사용자 요청 수입니다.

HTTP3xx

HTTP 상태 코드가 3xx 값(추가 작업 필요)인 최종 사용자 요청 수입니다.

HTTP4xx

HTTP 상태 코드가 4xx 값(클라이언트 오류)인 최종 사용자 요청 수입니다.

HTTP5xx

HTTP 상태 코드가 5xx 값(서버 오류)인 최종 사용자 요청 수입니다.

TotalBytes

모든 HTTP 메서드에 대한 모든 요청의 응답으로 CloudFront에서 최종 사용자에게 제공한 총 바이트 수입니다.

BytesFromMisses

요청 시 엣지 캐시에 없는 객체에 대해 최종 사용자에게 제공한 바이트 수입니다. 이 값은 오리진에서 CloudFront 엣지 캐시로 전송된 예상 정상 바이트 수입니다. 하지만 이미 엣지 캐시에 있지만 만료된 객체에 대한 요청은 제외됩니다.

캐시 통계 차트와 CloudFront 액세스 로그 데이터 간의 연결

다음 표에는 CloudFront 콘솔의 캐시 통계 차트가 CloudFront 액세스 로그의 값과 어떻게 연결되는지 보여줍니다. CloudFront 액세스 로그에 대한 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

Total Requests

이 차트에는 해당 기간 동안의 전체 HTTP 상태 코드(예: 200 또는 404)와 전체 메서드(예: `GET`, `HEAD` 또는 `POST`)의 총 요청 수가 표시됩니다. 이 차트에 표시된 총 요청 수는 동일 기간 동안의 액세스 로그 파일의 총 요청 수와 동일합니다.

Percentage of Viewer Requests by Result Type

이 차트에는 선택한 CloudFront 배포에 대한 적중, 누락 및 오류 수를 총 최종 사용자 요청 수의 백분율로 표시합니다.

- Hit – 객체가 CloudFront 엣지 캐시에서 제공되는 최종 사용자 요청입니다. 액세스 로그에서 이러한 요청은 `x-edge-response-result-type`의 값이 `Hit`인 요청입니다.
- Miss – 객체가 현재 엣지 캐시에 없어서 CloudFront가 객체를 오리진에서 가져와야 하는 최종 사용자 요청입니다. 액세스 로그에서 이러한 요청은 `x-edge-response-result-type`의 값이 `Miss`인 요청입니다.
- Error – 오류가 발생하여 CloudFront에서 객체를 제공하지 못한 최종 사용자 요청 수입니다. 액세스 로그에서 이러한 요청은 `x-edge-response-result-type`의 값이 `Error`, `LimitExceeded` 또는 `CapacityExceeded`인 요청입니다.

이 차트에는 엣지 캐시에 있지만 만료된 객체에 대한 적중 새로 고침 — 요청이 포함되지 않습니다. 액세스 로그에서 적중 새로 고침은 `x-edge-response-result-type`의 값이 `RefreshHit`인 요청입니다.

Bytes Transferred to Viewers

이 차트에는 두 가지 값이 표시됩니다.

- Total Bytes(총 바이트 수) – 모든 HTTP 메서드와 관련된 모든 요청에 대한 응답으로 CloudFront에서 최종 사용자에게 제공한 총 바이트 수입니다. CloudFront 액세스 로그에서 Total Bytes(총 바이트 수)는 동일 기간 동안 모든 요청에 대한 `sc-bytes` 열의 값 합계입니다.
- Bytes from Misses(누락 바이트 수) – 요청 시 엣지 캐시에 없는 객체에 대해 최종 사용자에게 제공한 바이트 수입니다. CloudFront 액세스 로그에서 Bytes from Misses(누락 바이트 수)는 `x-edge-result-type`의 값이 `Miss`인 요청에 대한 `sc-bytes` 열의 값 합계입니다. 이 값은 오리진에서 CloudFront 엣지 캐시로 전송된 예상 정상 바이트 수입니다. 하지만 이미 엣지 캐시에 있지만 만료된 객체에 대한 요청은 제외됩니다.

HTTP Status Codes

이 차트에는 HTTP 상태 코드에 의한 최종 사용자 요청이 표시됩니다. CloudFront 액세스 로그에서 상태 코드는 `sc-status` 열에 나타납니다.

- 2xx – 요청이 성공했습니다.
- 3xx – 추가 작업이 필요합니다. 예를 들어 301(영구 이동됨)은 요청된 객체가 다른 위치로 이동되었음을 의미합니다.
- 4xx – 명백하게 클라이언트로 인해 오류가 발생했습니다. 예를 들어 404(찾을 수 없음)는 클라이언트가 찾을 수 없는 객체를 요청했음을 의미합니다.
- 5xx – 오리진 서버에서 요청을 이행하지 않았습니다. 예를 들어 503(서비스를 사용할 수 없음)은 오리진 서버를 현재 사용할 수 없음을 의미합니다.

Percentage of GET Requests that Didn't Finish Downloading

이 차트에는 요청한 객체의 다운로드를 완료하지 않은 최종 사용자 `GET` 요청이 총 요청 수의 백분율로 표시됩니다. 일반적으로 다른 링크를 클릭하거나 브라우저를 닫는 등의 동작으로 최종 사용자가 다운로드를 취소하여 객체 다운로드가 완료되지 않습니다. CloudFront 액세스 로그에서 이러한 요청은 `sc-status` 열의 값이 `200`이며 `x-edge-result-type` 열의 값은 `Error`입니다.

CloudFront 인기 객체 보고서

Amazon CloudFront 콘솔은 이전 60일 내에서 지정된 날짜 범위 동안 배포에 대한 인기 있는 상위 50개 객체의 목록을 표시할 수 있습니다.

인기 객체 데이터 보고서는 CloudFront 액세스 로그와 동일한 소스에서 가져옵니다. 상위 50개 객체의 정확한 개수를 가져오기 위해 CloudFront는 자정부터 10분 간격으로 모든 객체에 대한 요청 수를 세고 다음 24시

간 동안 상위 150개 객체의 누계를 기록합니다. CloudFront는 60일 동안의 상위 150개 객체에 대한 일일 합계도 유지합니다. 목록 아래쪽 부근에서 객체가 목록에 새로 올라오거나 목록에서 누락되므로 이러한 객체의 합계는 근사치입니다. 150개 객체 목록에서 상위 50개 객체는 목록 내에서 순위가 올라가거나 내려갈 수 있지만, 목록에서 누락되는 일은 거의 드물며, 따라서 이러한 객체의 합계는 일반적으로 신뢰성이 더 높습니다.

어떤 객체가 상위 150개 객체 목록에서 누락된 후 하루가 지나서 다시 올라오는 경우 CloudFront는 해당 객체가 목록에서 누락되었던 기간 동안의 예상 요청 수를 추가합니다. 이 예상치는 객체가 해당 기간 동안 목록 아래쪽에 있을 때 받은 요청 수에 기초합니다. 객체가 당일 늦게 상위 50개 객체 범위 안으로 진입하는 경우 해당 객체가 상위 150개 객체에 포함되지 않았던 시간 동안 CloudFront에서 받은 요청 수의 예상치로 인해 일반적으로 인기 객체 보고서의 요청 수가 액세스 로그에서 해당 객체에 대해 표시되는 요청 수를 초과합니다.

Note

인기 객체 목록을 보기 위해 액세스 로깅을 활성화할 필요가 없습니다.

배포에 대한 인기 객체를 표시하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 탐색 창에서 Popular Objects(인기 객체)를 클릭합니다.
3. CloudFront 인기 객체 보고서 창의 시작일 및 종료일에서 인기 객체 목록을 표시할 날짜 범위를 선택합니다. 이전 60일 내에서 원하는 날짜 범위를 선택할 수 있습니다.

   날짜 및 시간이 협정 세계시(UTC)로 표시됩니다.
4. 배포 목록에서 인기 객체 목록을 표시할 배포를 선택합니다.
5. 업데이트를 클릭합니다.

주제

- 데이터를 CSV 형식으로 다운로드 (p. 375)
- 인기 객체 보고서 데이터와 CloudFront 액세스 로그 데이터 간의 연결 (p. 377)

데이터를 CSV 형식으로 다운로드

인기 객체 보고서를 CSV 형식으로 다운로드할 수 있습니다. 이 단원에서는 보고서를 다운로드하는 방법과 보고서의 값을 설명합니다.

인기 객체 보고서를 CSV 형식으로 다운로드하려면

1. 인기 객체 보고서를 표시한 상태에서 CSV를 클릭합니다.
2. 파일 이름 열기 대화 상자에서 파일을 열지 또는 저장할지 여부를 선택합니다.

보고서에 대한 정보

보고서의 처음 몇 줄에는 다음 정보가 포함됩니다.

버전

이 CSV 파일 형식의 버전입니다.

보고서

보고서의 이름입니다.

DistributionID

보고서가 실행된 배포의 ID입니다.

StartDateUTC

보고서 실행 기간에 해당하는 날짜 범위의 시작 일시를 협정 세계시(UTC)로 표시합니다.

EndDateUTC

보고서 실행 기간에 해당하는 날짜 범위의 종료 일시를 협정 세계시(UTC)로 표시합니다.

GeneratedTimeUTC

보고서를 실행 날짜 및 시간을 협정 세계시(UTC)로 표시합니다.

인기 객체 보고서의 데이터

이 보고서에는 다음 값이 포함됩니다.

DistributionID

보고서가 실행된 배포의 ID입니다.

FriendlyName

배포의 대체 도메인 이름(CNAME)입니다(있는 경우). 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.

객체

객체 URL의 마지막 500자입니다.

RequestCount

이 객체에 대한 총 요청 수입니다.

HitCount

CloudFront 엣지 캐시에서 객체가 제공되는 최종 사용자 요청 수입니다.

MissCount

현재 엣지 캐시에 객체가 없어서 CloudFront가 오리진에서 객체를 가져와야 하는 최종 사용자 요청 수입니다.

HitCountPct

`HitCount`의 값을 `RequestCount` 값의 백분율로 표시합니다.

BytesFromMisses

요청 시 객체가 엣지 캐시에 없는 경우 이 객체에 대해 최종 사용자에게 제공된 바이트 수입니다.

TotalBytes

모든 HTTP 메서드와 관련된 모든 요청에 대한 응답으로 CloudFront에서 이 객체에 대해 최종 사용자에게 제공한 총 바이트 수입니다.

IncompleteDownloadCount

이 객체에 대해 최종 사용자가 객체 다운로드를 시작했지만 완료하지 못한 최종 사용자 요청 수입니다.

HTTP2xx

HTTP 상태 코드가 2xx 값(성공)인 최종 사용자 요청 수입니다.

HTTP3xx

HTTP 상태 코드가 3xx 값(추가 작업 필요)인 최종 사용자 요청 수입니다.

HTTP4xx

HTTP 상태 코드가 4xx 값(클라이언트 오류)인 최종 사용자 요청 수입니다.

HTTP5xx

HTTP 상태 코드가 5xx 값(서버 오류)인 최종 사용자 요청 수입니다.

인기 객체 보고서 데이터와 CloudFront 액세스 로그 데이터 간의 연결

다음 목록은 CloudFront 콘솔의 인기 객체 보고서 값이 CloudFront 액세스 로그의 값과 어떻게 연결되는지 보여 줍니다. CloudFront 액세스 로그에 대한 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

URL

최종 사용자가 객체에 액세스하는 데 사용하는 마지막 500자입니다.

요청

객체에 대한 총 요청 수입니다. 이 값은 일반적으로 CloudFront 액세스 로그의 객체에 대한 `GET` 요청 수와 거의 같습니다.

Hits

CloudFront 엣지 캐시에서 객체가 제공된 최종 사용자 요청 수입니다. 액세스 로그에서 이러한 요청은 `x-edge-response-result-type`의 값이 `Hit`인 요청입니다.

Misses

엣지 캐시에 객체가 없어서 CloudFront가 오리진에서 객체를 가져왔던 최종 사용자 요청 수입니다. 액세스 로그에서 이러한 요청은 `x-edge-response-result-type`의 값이 `Miss`인 요청입니다.

Hit Ratio

Hits(적중 수) 열의 값을 요청 열 값의 백분율로 표시합니다.

Bytes from Misses

요청 시 엣지 캐시에 없는 객체에 대해 최종 사용자에게 제공한 바이트 수입니다. CloudFront 액세스 로그에서 Bytes from Misses(누락 바이트 수)는 `x-edge-result-type`의 값이 `Miss`인 요청에 대한 `sc-bytes` 열의 값 합계입니다.

Total Bytes

모든 HTTP 메서드에 대한 모든 요청의 응답으로 CloudFront에서 해당 객체에 대해 최종 사용자에게 제공한 총 바이트 수입니다. CloudFront 액세스 로그에서 Total Bytes(총 바이트 수)는 동일 기간 동안 모든 요청에 대한 `sc-bytes` 열의 값 합계입니다.

Incomplete Downloads

요청한 객체에 대한 다운로드를 완료하지 못한 최종 사용자 요청 수입니다. 일반적으로 다운로드가 완료되지 않은 이유는 다른 링크를 클릭하거나 브라우저를 닫는 등의 동작으로 최종 사용자가 취소했기 때문입니다. CloudFront 액세스 로그에서 이러한 요청은 `sc-status` 열의 값이 `200`이며 `x-edge-result-type` 열의 값은 `Error`입니다.

2xx

HTTP 상태 코드가 `2xx`, `Successful`인 요청 수입니다. CloudFront 액세스 로그에서 상태 코드는 `sc-status` 열에 나타납니다.

3xx

HTTP 상태 코드가 `3xx`, `Redirection`인 요청 수입니다. `3xx` 상태 코드는 추가 조치가 필요함을 나타냅니다. 예를 들어 301(영구 이동됨)은 요청된 객체가 다른 위치로 이동되었음을 의미합니다.

4xx

HTTP 상태 코드가 `4xx`, `Client Error`인 요청 수입니다. `4xx` 상태 코드는 명백하게 클라이언트로 인해 오류가 발생했음을 나타냅니다. 예를 들어 404(찾을 수 없음)는 클라이언트가 찾을 수 없는 객체를 요청했음을 의미합니다.

5xx

HTTP 상태 코드가 `5xx`, `Server Error`인 요청 수입니다. `5xx` 상태 코드는 오리진 서버에서 요청을 이행하지 않았음을 나타냅니다. 예를 들어 503(서비스를 사용할 수 없음)은 오리진 서버를 현재 사용할 수 없음을 의미합니다.

CloudFront 상위 참조자 보고서

CloudFront 콘솔에서는 지정된 배포에 대해 CloudFront에서 배포하고 있는 객체에 대한 대부분의 HTTP 및 HTTPS 요청이 시작된 웹 사이트의 상위 25개 도메인 목록을 표시할 수 있습니다. 이러한 상위 참조자는 검색 엔진, 객체에 직접 연결하는 기타 웹 사이트 또는 자체 웹 사이트일 수 있습니다. 예를 들어 http://example.com/index.html이 10개 그래픽과 연결되는 경우 example.com은 10개 그래픽 모두의 참조자입니다. 지난 60일 중 원하는 날짜 범위에 대해 상위 참조자 보고서를 표시할 수 있습니다.

> **Note**
>
> 사용자가 브라우저의 주소 표시줄에 URL을 직접 입력할 경우 다시 요청된 객체에 대한 참조자가 없습니다.

상위 참조자 보고서는 CloudFront 액세스 로그와 동일한 소스에서 가져옵니다. 상위 25개 참조자의 정확한 개수를 가져오기 위해 CloudFront는 자정부터 10분 간격으로 모든 객체에 대한 요청 수를 세고 상위 75개 참조자의 누계를 기록합니다. 목록 아래쪽 부근에서 참조자가 목록에 새로 올라오거나 목록에서 누락되므로 이러한 참조자의 합계는 근사치입니다. 75개 참조자 목록에서 상위 25개 객체는 목록 내에서 순위가 올라가거나 내려갈 수 있지만, 목록에서 누락되는 일은 거의 드물며, 따라서 이러한 참조자의 합계는 일반적으로 신뢰성이 더 높습니다.

> **Note**
>
> 상위 참조자 목록을 보기 위해 액세스 로깅을 활성화할 필요가 없습니다.

배포에 대한 상위 참조자를 표시하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 탐색 창에서 Top Referrers(상위 참조자)를 클릭합니다.
3. CloudFront 상위 참조자 보고서 창의 시작일 및 종료일에서 상위 참조자 목록을 표시할 날짜 범위를 선택합니다.

   날짜 및 시간이 협정 세계시(UTC)로 표시됩니다.
4. 배포 목록에서 상위 참조자 목록을 표시할 배포를 선택합니다.
5. 업데이트를 클릭합니다.

주제

- 데이터를 CSV 형식으로 다운로드 (p. 379)

- 상위 참조자 보고서 데이터와 CloudFront 액세스 로그 데이터 간의 연결 (p. 380)

데이터를 CSV 형식으로 다운로드

상위 참조자 보고서를 CSV 형식으로 다운로드할 수 있습니다. 이 단원에서는 보고서를 다운로드하는 방법과 보고서의 값을 설명합니다.

상위 참조자 보고서를 CSV 형식으로 다운로드하려면

1. 상위 참조자 보고서를 보는 동안 CSV를 클릭합니다.
2. 파일 이름 열기 대화 상자에서 파일을 열지 또는 저장할지 여부를 선택합니다.

보고서에 대한 정보

보고서의 처음 몇 줄에는 다음 정보가 포함됩니다.

버전

이 CSV 파일 형식의 버전입니다.

보고서

보고서의 이름입니다.

DistributionID

보고서가 실행된 배포의 ID 또는, 모든 배포에 대한 보고서를 실행한 경우 `ALL`입니다.

StartDateUTC

보고서 실행 기간에 해당하는 날짜 범위의 시작 일시를 협정 세계시(UTC)로 표시합니다.

EndDateUTC

보고서 실행 기간에 해당하는 날짜 범위의 종료 일시를 협정 세계시(UTC)로 표시합니다.

GeneratedTimeUTC

보고서를 실행 날짜 및 시간을 협정 세계시(UTC)로 표시합니다.

상위 참조자 보고서의 데이터

이 보고서에는 다음 값이 포함됩니다.

DistributionID

보고서가 실행된 배포의 ID 또는, 모든 배포에 대한 보고서를 실행한 경우 `ALL`입니다.

FriendlyName

배포의 대체 도메인 이름(CNAME)입니다(있는 경우). 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.

Referrer

참조자의 도메인 이름입니다.

`Referrer` 열의 도메인 이름에서 받은 총 요청 수입니다.

RequestsPct

지정된 기간 동안 참조가 제출한 요청 수를 총 요청 수의 백분율로 표시합니다.

상위 참조자 보고서 데이터와 CloudFront 액세스 로그 데이터 간의 연결

다음 목록은 CloudFront 콘솔의 상위 참조자 보고서 값이 CloudFront 액세스 로그의 값과 어떻게 연결되는지 보여 줍니다. CloudFront 액세스 로그에 대한 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

Referrer

참조자의 도메인 이름입니다. 액세스 로그에서 참조자는 `cs(Referer)` 열에 나열됩니다.

Request Count

참조자 열의 도메인 이름에서 받은 총 요청 수입니다. 이 값은 일반적으로 CloudFront 액세스 로그의 참조자에서 받은 `GET` 요청 수와 거의 같습니다.

요청 %

지정된 기간 동안 참조가 제출한 요청 수를 총 요청 수의 백분율로 표시합니다. 참조자 수가 25개 이상이면 이 테이블의 데이터에 따라 Request %(요청 %)를 계산할 수 없습니다. Request Count(요청 수) 열에 지정된 기간 동안의 모든 요청이 다 포함되지 않기 때문입니다.

CloudFront 사용 보고서

Amazon CloudFront 콘솔은 사용 보고서 데이터의 하위 집합을 기준으로 하는 CloudFront 사용량을 그래픽으로 표시할 수 있습니다. 한 시간 또는 하루 기준의 데이터 지점을 포함해 지난 60일 동안 지정된 날짜 범위에 대한 차트를 표시할 수 있습니다. 일반적으로 네 시간 전까지 CloudFront에서 수신한 요청에 대한 데이터를 볼 수 있지만, 이따금씩 데이터가 24시간 가량 지연될 수 있습니다.

자세한 내용은 사용 차트가 CloudFront 사용 보고서의 데이터와 연결되는 방법 (p. 383) 단원을 참조하십시오.

CloudFront 사용 차트를 표시하는 방법

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 탐색 창에서 사용 보고서를 클릭합니다.
3. CloudFront 사용 보고서 창의 시작일 및 종료일 필드에서 사용 차트를 표시할 날짜 범위를 선택합니다. 사용 가능한 범위는 세부 수준에 선택한 값에 따라 다릅니다.

   - 매일 — 하루에 데이터 지점 하나씩 포함하는 차트를 표시하려면 이전 60일에 해당하는 날짜 범위를 선택합니다.
   - 매 시간 — 한 시간에 데이터 지점 하나씩 포함하는 차트를 표시하려면 이전 60일 내에서 날짜 범위를 최대 14일까지 선택합니다.

   날짜 및 시간이 협정 세계시(UTC)로 표시됩니다.
4. 세부 수준의 경우 차트에 하루에 데이터 지점 하나씩 표시할지, 한 시간에 데이터 지점 하나씩 표시할지 지정합니다. 날짜 범위를 14일보다 길게 지정할 경우 한 시간에 데이터 지점 하나를 지정하는 옵션을 사용할 수 없습니다.
5. Billing Region(결제 리전)에서 표시할 데이터가 있는 CloudFront 결제 리전을 선택하거나 All Regions(모든 리전)를 선택합니다. 사용 차트에는 CloudFront가 지정된 지역의 엣지에서 처리하는 요청에 대한 데

이터가 포함됩니다. CloudFront가 요청을 처리하는 지역은 사용자의 위치에 해당되거나 해당되지 않을 수 있습니다.

배포에 대한 가격 등급에 포함된 지역만 선택하세요. 그렇지 않으면 사용 차트에 데이터가 포함되어 있지 않을 가능성이 큽니다. 예를 들어 배포에 대해 가격 등급 200을 선택한 경우 남미 및 오스트레일리아 결제 지역이 포함되지 않으므로, CloudFront에서는 일반적으로 이 지역의 요청을 처리하지 않습니다. 가격 등급에 대한 자세한 내용은 CloudFront 배포의 가격 등급 선택 (p. 9) 단원을 참조하십시오.

6. 배포 목록에서 사용 차트에 데이터를 표시할 배포를 선택합니다.
   - 개별 웹 배포 — 차트에 선택한 CloudFront 배포에 대한 데이터가 표시됩니다. 배포 목록은 배포에 대한 ID와 대체 도메인 이름(CNAME)이 있는 경우 이를 표시합니다. 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.
   - All Web Distributions(excludes deleted)(모든 웹 배포(삭제된 배포 제외)) — 이 차트에는 사용자가 삭제한 웹 배포를 제외한 현재 AWS 계정과 연결된 모든 웹 배포에 대해 합산된 데이터가 표시됩니다.
   - All Deleted Distributions(삭제된 모든 배포) — 차트에 현재 AWS 계정과 연결되어 있고 지난 60일 동안 삭제된 모든 웹 배포에 대해 합산된 데이터가 표시됩니다.
7. Update Graphs(그래프 업데이트)를 클릭합니다.
8. 차트 안에서 일별 또는 시간별 데이터를 보려면 마우스 포인터를 데이터 지점 위로 이동합니다.
9. 전송된 데이터를 보여 주는 차트의 경우 각 차트에 대해 수직 눈금을 기가바이트, 메가바이트 또는 킬로바이트로 변경할 수 있습니다.

주제

- 데이터를 CSV 형식으로 다운로드 (p. 381)
- 사용 차트가 CloudFront 사용 보고서의 데이터와 연결되는 방법 (p. 383)

데이터를 CSV 형식으로 다운로드

사용 보고서를 CSV 형식으로 다운로드할 수 있습니다. 이 단원에서는 보고서를 다운로드하는 방법과 보고서의 값을 설명합니다.

사용 보고서를 CSV 형식으로 다운로드하려면

1. 사용 보고서를 표시한 상태에서 CSV를 클릭합니다.
2. 파일 이름 열기 대화 상자에서 파일을 열지 또는 저장할지 여부를 선택합니다.

보고서에 대한 정보

보고서의 처음 몇 줄에는 다음 정보가 포함됩니다.

버전

이 CSV 파일 형식의 버전입니다.

보고서

보고서의 이름입니다.

DistributionID

보고서를 실행한 배포의 ID로, 모든 배포에 대한 보고서를 실행한 경우 `ALL`, 또는 삭제된 모든 배포에 대한 보고서를 실행한 경우 `ALL_DELETED`입니다.

StartDateUTC

보고서 실행 기간에 해당하는 날짜 범위의 시작 일시를 협정 세계시(UTC)로 표시합니다.

EndDateUTC

보고서 실행 기간에 해당하는 날짜 범위의 종료 일시를 협정 세계시(UTC)로 표시합니다.

GeneratedTimeUTC

보고서를 실행 날짜 및 시간을 협정 세계시(UTC)로 표시합니다.

Granularity

보고서의 각 줄이 한 시간을 나타내는지 하루를 나타내는지 지정합니다.

BillingRegion

최종 사용자 요청이 시작된 대륙, 또는 모든 결제 리전에 대한 보고서를 다운로드하도록 선택한 경우 `ALL`입니다.

사용 보고서의 데이터

이 보고서에는 다음 값이 포함됩니다.

DistributionID

보고서를 실행한 배포의 ID로, 모든 배포에 대한 보고서를 실행한 경우 `ALL`, 또는 삭제된 모든 배포에 대한 보고서를 실행한 경우 `ALL_DELETED`입니다.

FriendlyName

배포의 대체 도메인 이름(CNAME)입니다(있는 경우). 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.

BillingRegion

보고서가 실행된 CloudFront 결제 리전, 또는 `ALL`입니다.

TimeBucket

데이터가 적용되는 시간 또는 날짜를 협정 세계시(UTC)로 표시합니다.

HTTP

지정된 CloudFront 배포에 대한 매 시간 간격 동안 선택한 리전의 엣지 위치에서 CloudFront가 응답한 HTTP 요청 수입니다. 값은 다음과 같습니다.

- CloudFront에서 데이터를 사용자에게 전송하게 하는 `GET` 및 `HEAD` 요청 수
- CloudFront에서 데이터를 오리진으로 전송하게 하는 `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청 수

HTTPS

지정된 CloudFront 배포에 대한 매 시간 간격 동안 선택한 리전의 엣지 로케이션에서 CloudFront가 응답한 HTTPS 요청 수입니다. 값은 다음과 같습니다.

- CloudFront에서 데이터를 사용자에게 전송하게 하는 `GET` 및 `HEAD` 요청 수
- CloudFront에서 데이터를 오리진으로 전송하게 하는 `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청 수

HTTPBytes

지정된 CloudFront 배포에 대해 일정 기간 동안 선택된 결제 리전의 CloudFront 엣지 로케이션에서 HTTP를 통해 전송된 총 데이터 양입니다. 값은 다음과 같습니다.

- `GET` 및 `HEAD` 요청에 대한 응답으로 CloudFront에서 사용자에게 전송된 데이터
- `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대해 CloudFront에서 오리진으로 전송된 데이터

- DELETE, OPTIONS, PATCH, POST 및 PUT 요청에 대한 응답으로 CloudFront에서 사용자에게 전송된 데이터

HTTPSBytes

지정된 CloudFront 배포에 대해 일정 기간 동안 선택된 결제 리전의 CloudFront 엣지 로케이션에서 HTTPS를 통해 전송된 총 데이터 양입니다. 값은 다음과 같습니다.

- `GET` 및 `HEAD` 요청에 대한 응답으로 CloudFront에서 사용자에게 전송된 데이터
- `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대해 CloudFront에서 오리진으로 전송된 데이터
- DELETE, OPTIONS, PATCH, POST 및 PUT 요청에 대한 응답으로 CloudFront에서 사용자에게 전송된 데이터

BytesIn

지정된 CloudFront 배포에 대해 각 기간 동안 선택한 리전의 `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대해 CloudFront에서 오리진으로 전송된 총 데이터 양입니다.

BytesOut

지정된 CloudFront 배포에 대해 일정 기간 동안 HTTP 및 HTTPS를 통해 CloudFront에서 선택된 결제 리전의 사용자에게 전송된 총 데이터 양입니다. 값은 다음과 같습니다.

- `GET` 및 `HEAD` 요청에 대한 응답으로 CloudFront에서 사용자에게 전송된 데이터
- `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대한 응답으로 CloudFront에서 사용자에게 전송된 데이터

사용 차트가 CloudFront 사용 보고서의 데이터와 연결되는 방법

다음 목록은 CloudFront 콘솔의 사용 차트가 CloudFront 사용 보고서의 [Usage Type] 열의 값과 어떻게 연결되는지 보여 줍니다.

주제

- Number of Requests (p. 383)
- Data Transferred by Protocol (p. 384)
- Data Transferred by Destination (p. 384)

Number of Requests

이 차트는 지정된 CloudFront 배포의 각 시간 간격에서 선택한 리전에 있는 엣지 위치에서 CloudFront이 응답하는 요청의 총 수를 프로토콜(HTTP 또는 HTTPS)과 유형(정적, 동적 또는 프록시)으로 구분하여 보여줍니다.

HTTP 요청 수

- *region*-Requests-HTTP-Static: TTL이 3600초보다 크거나 같은 객체에 서비스되는 HTTP `GET` 및 `HEAD` 요청의 수
- *region*-Requests-HTTP-Dynamic: TTL이 3600초보다 작은 객체에 서비스되는 HTTP `GET` 및 `HEAD` 요청의 수
- *region*-Requests-HTTP-Proxy: CloudFront가 오리진으로 전달하는 HTTP `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청의 수

HTTPS 요청 수

- *region*-Requests-HTTPS-Static: TTL이 3600초보다 크거나 같은 객체에 서비스되는 HTTPS `GET` 및 `HEAD` 요청의 수
- *region*-Requests-HTTPS-Dynamic: TTL이 3600초보다 작은 객체에 서비스되는 HTTPS `GET` 및 `HEAD` 요청의 수

- *region*-Requests-HTTPS-Proxy: CloudFront가 오리진으로 전달하는 HTTPS `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청의 수

Data Transferred by Protocol

이 차트는 프로토콜(HTTP 또는 HTTPS), 유형(정적, 동적 또는 프록시) 및 대상(사용자 또는 원본)으로 구분하여 지정된 CloudFront 배포의 각 시간 간격 동안 선택한 리전의 CloudFront 엣지 위치에서 전송된 데이터의 총 양을 보여줍니다.

HTTP를 통해 전송되는 데이터

- *region*-Out-Bytes-HTTP-Static: TTL이 3600초보다 크거나 같은 객체에 대해 HTTP를 통해 서비스되는 바이트
- *region*-Out-Bytes-HTTP-Dynamic: TTL이 3600초보다 작은 객체에 대해 HTTP를 통해 서비스되는 바이트
- *region*-Out-Bytes-HTTP-Proxy: `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대한 응답으로 HTTP를 통해 CloudFront에서 최종 사용자에게 반환되는 바이트
- *region*-Out-OBytes-HTTP-Proxy: `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대한 응답으로 HTTP를 통해 CloudFront 엣지 로케이션에서 오리진으로 전송된 총 바이트 수

HTTPS를 통해 전송되는 데이터

- *region*-Out-Bytes-HTTPS-Static: TTL이 3600초보다 크거나 같은 객체에 대해 HTTPS를 통해 서비스되는 바이트
- *region*-Out-Bytes-HTTPS-Dynamic: TTL이 3600초보다 작은 객체에 대해 HTTPS를 통해 서비스되는 바이트
- *region*-Out-Bytes-HTTPS-Proxy: `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대한 응답으로 HTTPS를 통해 CloudFront에서 최종 사용자에게 반환되는 바이트
- *region*-Out-OBytes-HTTPS-Proxy: `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대한 응답으로 HTTPS를 통해 CloudFront 엣지 로케이션에서 전송된 총 바이트 수

Data Transferred by Destination

이 차트는 지정된 CloudFront 배포에 대해 각 시간 간격 동안 선택한 리전의 CloudFront 엣지 로케이션에서 전송된 데이터 총 양을 대상(사용자 또는 오리진), 프로토콜(HTTP 또는 HTTPS) 및 유형(정적, 동적 또는 프록시)으로 구분하여 보여줍니다.

CloudFront에서 사용자에게 전송된 데이터

- *region*-Out-Bytes-HTTP-Static: TTL이 3600초보다 크거나 같은 객체에 대해 HTTP를 통해 서비스되는 바이트
- *region*-Out-Bytes-HTTPS-Static: TTL이 3600초보다 크거나 같은 객체에 대해 HTTPS를 통해 서비스되는 바이트
- *region*-Out-Bytes-HTTP-Dynamic: TTL이 3600초보다 작은 객체에 대해 HTTP를 통해 서비스되는 바이트
- *region*-Out-Bytes-HTTPS-Dynamic: TTL이 3600초보다 작은 객체에 대해 HTTPS를 통해 서비스되는 바이트
- *region*-Out-Bytes-HTTP-Proxy: `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대한 응답으로 HTTP를 통해 CloudFront에서 최종 사용자에게 반환되는 바이트
- *region*-Out-Bytes-HTTPS-Proxy: `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대한 응답으로 HTTPS를 통해 CloudFront에서 최종 사용자에게 반환되는 바이트

CloudFront에서 오리진에 전송되는 데이터

- *region*-Out-OBytes-HTTP-Proxy: `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대한 응답으로 HTTP를 통해 CloudFront 엣지 로케이션에서 오리진으로 전송된 총 바이트 수

- *region*-Out-OBytes-HTTPS-Proxy: `DELETE`, `OPTIONS`, `PATCH`, `POST` 및 `PUT` 요청에 대한 응답으로 HTTPS를 통해 CloudFront 엣지 로케이션에서 전송된 총 바이트 수

CloudFront Viewers Reports

CloudFront 콘솔에는 콘텐츠에 액세스하는 최종 사용자(일반 웹 브라우저)와 물리적 디바이스(데스크톱 컴퓨터, 모바일 디바이스)에 대한 네 가지 보고서를 표시할 수 있습니다.

- 디바이스 – 사용자가 콘텐츠에 액세스하는 데 자주 사용하는 디바이스의 유형(예: 데스크톱 또는 모바일)입니다.
- 브라우저 – 사용자가 콘텐츠에 액세스할 때 자주 사용하는 브라우저의 이름이나 버전(예: Chrome 또는 Firefox)을 표시합니다. 이 보고서에는 상위 10개 브라우저가 나열됩니다.
- 운영 체제 – 최종 사용자가 콘텐츠에 액세스할 때 자주 실행하는 운영 체제의 이름이나 버전(예: Linux, Mac OS X 또는 Windows)을 표시합니다. 이 보고서에는 상위 10개 운영 체제가 나열됩니다.
- 위치 – 콘텐츠에 자주 액세스하는 최종 사용자의 국가별 또는 미국 주/지역별 위치를 표시합니다. 이 보고서에는 상위 50개 국가 또는 미국 주/지역이 나열됩니다.

이전 60일 내에서 원하는 날짜 범위에 대해 네 가지 최종 사용자 보고서 모두를 표시할 수 있습니다. Locations 보고서의 경우 이전 60일 내에서 최장 14일까지의 날짜 범위 동안 매 시간별 데이터 지점을 포함하는 보고서를 표시할 수도 있습니다.

> **Note**
>
> 최종 사용자 차트 및 보고서를 보기 위해 액세스 로깅을 활성화할 필요는 없습니다.

주제

- 최종 사용자 차트 및 보고서 표시 (p. 385)
- 데이터를 CSV 형식으로 다운로드 (p. 386)
- Locations 보고서 데이터를 CloudFront 액세스 로그 데이터와 연결 (p. 391)

최종 사용자 차트 및 보고서 표시

CloudFront 최종 사용자 차트 및 보고서를 표시하려면 다음 절차를 수행합니다.

CloudFront 최종 사용자 차트 및 보고서를 표시하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 탐색 창에서 Viewers(최종 사용자)를 클릭합니다.
3. CloudFront 최종 사용자 창의 시작일 및 종료일에서 최종 사용자 차트 및 보고서를 표시할 날짜 범위를 선택합니다.

   Locations 차트의 경우 사용 가능한 범위는 세부 수준에 선택한 값에 따라 다릅니다.

   - 매일 – 하루에 데이터 지점 하나씩 포함하는 차트를 표시하려면 이전 60일에 해당하는 날짜 범위를 선택합니다.
   - 매 시간 – 한 시간에 데이터 지점 하나씩 포함하는 차트를 표시하려면 이전 60일 내에서 날짜 범위를 최대 14일까지 선택합니다.

   날짜 및 시간이 협정 세계시(UTC)로 표시됩니다.
4. (Browsers 및 Operating Systems 차트에만 해당) 그룹화에는 이름별로(Chrome, Firefox) 또는 이름 및 버전별로(Chrome 40.0, Firefox 35.0) 브라우저와 운영 체제를 그룹화할지 여부를 지정합니다.

5. (Locations 차트에만 해당) 세부 수준에는 차트에 하루에 데이터 지점 하나씩 표시할지, 한 시간에 데이터 지점 하나씩 표시할지 지정합니다. 날짜 범위를 14일보다 길게 지정할 경우 한 시간에 데이터 지점 하나를 지정하는 옵션을 사용할 수 없습니다.
6. (Locations에만 해당) 세부 정보에는 국가 또는 미국 주별 상위 위치를 표시할지 여부를 지정합니다.
7. 배포 목록에서 사용 차트에 데이터를 표시할 배포를 선택합니다.
   - 개별 웹 배포 – 이 차트에는 선택한 CloudFront 웹 배포에 대한 데이터가 표시됩니다. 배포 목록에는 배포에 대한 ID와 대체 도메인 이름(CNAME)이 표시됩니다(있는 경우). 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.
   - All Web Distributions(excludes deleted) – (모든 웹 배포(삭제된 배포 제외)) - 이 차트에는 사용자가 삭제한 웹 배포를 제외한 현재 AWS 계정과 연결된 모든 웹 배포에 대해 합산된 데이터가 표시됩니다.
8. 업데이트를 클릭합니다.
9. 차트 안에서 일별 또는 시간별 데이터를 보려면 마우스 포인터를 데이터 지점 위로 이동합니다.

데이터를 CSV 형식으로 다운로드

각각의 최종 사용자 보고서를 CSV 형식으로 다운로드할 수 있습니다. 이 단원에서는 보고서를 다운로드하는 방법과 보고서의 값을 설명합니다.

최종 사용자 보고서를 CSV 형식으로 다운로드하려면

1. 최종 사용자 보고서를 표시한 상태에서 CSV를 클릭합니다.
2. 디바이스 또는 Devices Trends(디바이스 추세) 같은, 다운로드할 데이터를 선택합니다.
3. 파일 이름 열기 대화 상자에서 파일을 열지 또는 저장할지 여부를 선택합니다.

주제

- 보고서에 대한 정보 (p. 386)
- Devices 보고서 (p. 387)
- Device Trends 보고서 (p. 387)
- Browsers 보고서 (p. 388)
- Browser Trends 보고서 (p. 388)
- Operating Systems 보고서 (p. 389)
- Operating System Trends 보고서 (p. 389)
- Locations 보고서 (p. 390)
- Location Trends Report (p. 390)

보고서에 대한 정보

각 보고서의 처음 몇 줄에는 다음 정보가 포함됩니다.

버전

이 CSV 파일 형식의 버전입니다.

보고서

보고서의 이름입니다.

DistributionID

보고서가 실행된 배포의 ID 또는, 모든 웹 배포에 대한 보고서를 실행한 경우 `ALL`입니다.

StartDateUTC

보고서 실행 기간에 해당하는 날짜 범위의 시작 일시를 협정 세계시(UTC)로 표시합니다.

EndDateUTC

보고서 실행 기간에 해당하는 날짜 범위의 종료 일시를 협정 세계시(UTC)로 표시합니다.

GeneratedTimeUTC

보고서를 실행 날짜 및 시간을 협정 세계시(UTC)로 표시합니다.

그룹화(Browsers 및 Operating Systems 보고서에만 해당)

데이터를 이름별로 그룹화할지 브라우저나 운영 체제의 이름 및 버전별로 그룹화할지 여부를 지정합니다.

Granularity

보고서의 각 줄이 한 시간을 나타내는지 하루를 나타내는지 지정합니다.

세부 정보(Locations 보고서에만 해당)

요청을 국가별로 나열할지 또는 미국 주별로 나열할지 여부를 지정합니다.

Devices 보고서

이 보고서에는 다음 값이 포함됩니다.

DistributionID

보고서가 실행된 배포의 ID 또는, 모든 배포에 대한 보고서를 실행한 경우 `ALL`입니다.

FriendlyName

배포의 대체 도메인 이름(CNAME)입니다(있는 경우). 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.

요청

CloudFront가 각 디바이스 유형에서 받은 요청 수입니다.

RequestsPct

CloudFront가 각 디바이스 유형에서 받은 요청 수를 CloudFront가 모든 디바이스에서 받은 총 요청 수의 백분율로 표시합니다.

Device Trends 보고서

이 보고서에는 다음 값이 포함됩니다.

DistributionID

보고서가 실행된 배포의 ID 또는, 모든 배포에 대한 보고서를 실행한 경우 `ALL`입니다.

FriendlyName

배포의 대체 도메인 이름(CNAME)입니다(있는 경우). 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.

TimeBucket

데이터가 적용되는 시간 또는 날짜를 협정 세계시(UTC)로 표시합니다.

Desktop

기간 중에 CloudFront가 데스크톱 컴퓨터에서 받은 요청 수입니다.

모바일

기간 중에 CloudFront가 모바일 디바이스에서 받은 요청 수입니다. 모바일 디바이스에는 태블릿과 휴대폰이 포함될 수 있습니다. CloudFront가 요청이 모바일 디바이스나 태블릿에서 시작되었는지 확인할 수 없는 경우 해당 요청이 `Mobile` 열의 개수에 포함됩니다.

Smart-TV

CloudFront가 해당 기간 동안 스마트 TV에서 받은 요청 수입니다.

Tablet

해당 기간 동안 CloudFront가 태블릿에서 받은 요청 수입니다. CloudFront가 요청이 모바일 디바이스나 태블릿에서 시작되었는지 확인할 수 없는 경우 해당 요청이 `Mobile` 열의 개수에 포함됩니다.

알 수 없음

`User-Agent` HTTP 헤더 값이 표준 디바이스 유형(예: `Desktop` 또는 `Mobile`) 중 하나와 연결되지 않은 요청입니다.

Empty

해당 기간 동안 CloudFront가 받은 요청 중에서 HTTP `User-Agent` 헤더에 값이 포함되지 않은 요청의 수입니다.

Browsers 보고서

이 보고서에는 다음 값이 포함됩니다.

DistributionID

보고서가 실행된 배포의 ID 또는, 모든 배포에 대한 보고서를 실행한 경우 `ALL`입니다.

FriendlyName

배포의 대체 도메인 이름(CNAME)입니다(있는 경우). 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.

그룹

`Grouping` 값에 따라, CloudFront에서 요청을 받은 브라우저 또는 브라우저 및 버전입니다. 브라우저 이름 외에도 다음과 같은 값을 사용할 수 있습니다.

- Bot/Crawler – 기본적으로 콘텐츠를 인덱싱하는 검색 엔진의 요청입니다.
- Empty – `User-Agent` HTTP 헤더의 값이 비어 있는 요청입니다.
- Other – CloudFront에서 식별되었지만 잘 사용되지 않는 브라우저입니다. 처음 9개 값 중에 `Bot/Crawler`, `Empty` 및/또는 `Unknown`이 표시되지 않으면 이러한 값이 `Other`에도 포함됩니다.
- Unknown – `User-Agent` HTTP 헤더의 값이 표준 브라우저와 연결되지 않은 요청입니다. 이 범주에 속하는 대부분의 요청은 사용자 지정 애플리케이션이나 스크립트에서 들어옵니다.

요청

CloudFront가 각 브라우저 유형에서 받은 요청 수입니다.

RequestsPct

CloudFront가 각 브라우저 유형에서 받은 요청 수를 해당 기간 동안 CloudFront가 받은 총 요청 수의 백분율로 표시합니다.

Browser Trends 보고서

이 보고서에는 다음 값이 포함됩니다.

DistributionID

보고서가 실행된 배포의 ID 또는, 모든 배포에 대한 보고서를 실행한 경우 `ALL`입니다.

FriendlyName

배포의 대체 도메인 이름(CNAME)입니다(있는 경우). 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.

TimeBucket

데이터가 적용되는 시간 또는 날짜를 협정 세계시(UTC)로 표시합니다.

(브라우저)

보고서의 나머지 열에는 `Grouping` 값에 따라 브라우저 또는 브라우저 및 버전이 나열됩니다. 브라우저 이름 외에도 다음과 같은 값을 사용할 수 있습니다.

- Bot/Crawler – 기본적으로 콘텐츠를 인덱싱하는 검색 엔진의 요청입니다.
- Empty – `User-Agent` HTTP 헤더의 값이 비어 있는 요청입니다.
- Other – CloudFront에서 식별되었지만 잘 사용되지 않는 브라우저입니다. 처음 9개 값 중에 `Bot/Crawler`, `Empty` 및/또는 `Unknown`이 표시되지 않으면 이러한 값이 `Other`에도 포함됩니다.
- Unknown – `User-Agent` HTTP 헤더의 값이 표준 브라우저와 연결되지 않은 요청입니다. 이 범주에 속하는 대부분의 요청은 사용자 지정 애플리케이션이나 스크립트에서 들어옵니다.

Operating Systems 보고서

이 보고서에는 다음 값이 포함됩니다.

DistributionID

보고서가 실행된 배포의 ID 또는, 모든 배포에 대한 보고서를 실행한 경우 `ALL`입니다.

FriendlyName

배포의 대체 도메인 이름(CNAME)입니다(있는 경우). 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.

그룹

`Grouping` 값에 따라, CloudFront에서 요청을 받은 운영 체제 또는 운영 체제 및 버전입니다. 운영 체제 이름 외에도 다음과 같은 값을 사용할 수 있습니다.

- Bot/Crawler – 기본적으로 콘텐츠를 인덱싱하는 검색 엔진의 요청입니다.
- Empty – `User-Agent` HTTP 헤더의 값이 비어 있는 요청입니다.
- Other – CloudFront에서 식별되었지만 잘 사용되지 않는 운영 체제입니다. 처음 9개 값 중에 `Bot/Crawler`, `Empty` 및/또는 `Unknown`이 표시되지 않으면 이러한 값이 `Other`에도 포함됩니다.
- Unknown – `User-Agent` HTTP 헤더의 값이 표준 브라우저와 연결되지 않은 요청입니다. 이 범주에 속하는 대부분의 요청은 사용자 지정 애플리케이션이나 스크립트에서 들어옵니다.

요청

CloudFront가 각 운영 체제 유형에서 받은 요청 수입니다.

RequestsPct

CloudFront가 각 운영 체제 유형에서 받은 요청 수를 해당 기간 동안 CloudFront가 받은 총 요청 수의 백분율로 표시합니다.

Operating System Trends 보고서

이 보고서에는 다음 값이 포함됩니다.

DistributionID

보고서가 실행된 배포의 ID 또는, 모든 배포에 대한 보고서를 실행한 경우 `ALL`입니다.

FriendlyName

배포의 대체 도메인 이름(CNAME)입니다(있는 경우). 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.

TimeBucket

데이터가 적용되는 시간 또는 날짜를 협정 세계시(UTC)로 표시합니다.

(운영 체제)

보고서의 나머지 열에는 `Grouping` 값에 따라 운영 체제 또는 운영 체제 및 버전이 나열됩니다. 운영 체제 이름 외에도 다음과 같은 값을 사용할 수 있습니다.

- Bot/Crawler – 기본적으로 콘텐츠를 인덱싱하는 검색 엔진의 요청입니다.
- Empty – `User-Agent` HTTP 헤더의 값이 비어 있는 요청입니다.
- Other – CloudFront에서 식별되었지만 잘 사용되지 않는 운영 체제입니다. 처음 9개 값 중에 `Bot/Crawler`, `Empty` 및/또는 `Unknown`이 표시되지 않으면 이러한 값이 `Other`에도 포함됩니다.
- Unknown – `User-Agent` HTTP 헤더에 운영 체제가 지정되지 않은 요청입니다.

Locations 보고서

이 보고서에는 다음 값이 포함됩니다.

DistributionID

보고서가 실행된 배포의 ID 또는, 모든 배포에 대한 보고서를 실행한 경우 `ALL`입니다.

FriendlyName

배포의 대체 도메인 이름(CNAME)입니다(있는 경우). 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.

LocationCode

CloudFront에서 요청을 받은 위치의 약어입니다. 사용 가능한 값에 대한 자세한 내용은 Locations 보고서 데이터를 CloudFront 액세스 로그 데이터와 연결 (p. 391)의 Location 설명을 참조하십시오.

LocationName

CloudFront에서 요청을 받은 위치의 이름입니다.

요청

CloudFront가 각 위치에서 받은 요청 수입니다.

RequestsPct

CloudFront가 각 위치에서 받은 요청 수를 해당 기간 동안 CloudFront가 모든 위치에서 받은 총 요청 수의 백분율로 표시합니다.

TotalBytes

CloudFront가 지정된 배포 및 기간 동안 이 국가나 주의 최종 사용자에게 제공한 바이트 수입니다.

Location Trends Report

이 보고서에는 다음 값이 포함됩니다.

DistributionID

보고서가 실행된 배포의 ID 또는, 모든 배포에 대한 보고서를 실행한 경우 `ALL`입니다.

FriendlyName

배포의 대체 도메인 이름(CNAME)입니다(있는 경우). 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.

TimeBucket

데이터가 적용되는 시간 또는 날짜를 협정 세계시(UTC)로 표시합니다.

(위치)

보고서의 나머지 열에는 CloudFront가 요청을 받은 위치가 나열됩니다. 사용 가능한 값에 대한 자세한 내용은 Locations 보고서 데이터를 CloudFront 액세스 로그 데이터와 연결 (p. 391)의 Location 설명을 참조하십시오.

Locations 보고서 데이터를 CloudFront 액세스 로그 데이터와 연결

다음 목록은 CloudFront 콘솔의 Locations 보고서 데이터가 CloudFront 액세스 로그의 값과 어떻게 연결되는지 보여 줍니다. CloudFront 액세스 로그에 대한 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

위치

최종 사용자가 위치한 국가 또는 미국 주입니다. 액세스 로그에서 `c-ip` 열에는 최종 사용자가 실행 중인 디바이스의 IP 주소가 포함됩니다. 지리적 위치 데이터를 사용하여 IP 주소에 따라 디바이스의 지리적 위치를 식별합니다.

국가별로 위치 보고서를 표시하는 경우 국가 목록은 ISO 3166-2, Codes for the representation of names of countries and their subdivisions – Part 2: Country subdivision code에 기초합니다. 국가 목록에는 다음 추가 값이 포함됩니다.

- Anonymous Proxy(익명 프록시) – 익명 프록시에서 시작된 요청입니다.
- Satellite Provider(위성 공급자) – 여러 국가에 인터넷 서비스를 제공하는 위성 공급자로부터 시작된 요청입니다. 사용자가 사기 위험이 높은 국가에 있을 수 있습니다.
- Europe (Unknown)(유럽(알 수 없음)) – 여러 유럽 국가에서 사용되는 블록의 IP에서 시작된 요청입니다. 요청이 시작된 국가를 결정할 수 없습니다. CloudFront에서는 기본값으로 Europe (Unknown)(유럽(알 수 없음))을 사용합니다.
- Asia/Pacific (Unknown)(아시아/태평양(알 수 없음)) – 여러 아시아/태평양 리전 국가에서 사용되는 블록의 IP에서 시작된 요청입니다. 요청이 시작된 국가를 결정할 수 없습니다. CloudFront에서는 기본값으로 Asia/Pacific (Unknown)(아시아/태평양(알 수 없음))을 사용합니다.

미국 주별로 위치 보고서를 표시하는 경우 보고서에 미국 지역과 미군 주둔 리전이 포함될 수 있습니다.

> **Note**
>
> CloudFront에서 사용자 위치를 확인할 수 없는 경우에는 최종 사용자 보고서에 위치가 "알 수 없음"으로 표시됩니다.

Request Count

지정된 배포 및 기간 동안 최종 사용자가 위치한 국가나 미국 주에서 받은 총 요청 수입니다. 일반적으로 이 값은 CloudFront 액세스 로그의 해당 국가 또는 주의 IP 주소에서 받은 `GET` 요청 수와 거의 같습니다.

요청 %

세부 정보에 선택한 값에 따라 다음 중 하나입니다.

- 국가 – 이 국가의 요청을 총 요청 수의 백분율로 표시합니다.

- 미국 – 이 주의 요청을 총 요청 수의 백분율로 표시합니다.

  50개 이상의 국가에서 요청을 받은 경우 이 테이블의 데이터에 따라 Request %(요청 %)를 계산할 수 없습니다. Request Count(요청 수) 열에 지정된 기간 동안의 모든 요청이 다 포함되지 않기 때문입니다.

바이트

CloudFront가 지정된 배포 및 기간 동안 이 국가나 주의 최종 사용자에게 제공한 바이트 수입니다. 이 열의 데이터 표시를 KB, MB 또는 GB로 변경하려면 열 머리글의 링크를 클릭합니다.

AWS Config를 통해 구성 변경 추적

AWS Config를 사용해 CloudFront 배포 설정 변경에 대한 구성 변경 사항을 기록할 수 있습니다. 예를 들어, 배포 상태, 가격 등급, 오리진, 지리적 제한 설정, Lambda@Edge 구성에 대한 변경 사항을 캡처할 수 있습니다.

Note

AWS Config에서는 CloudFront 배포에 대한 키-값 태그를 기록하지 않습니다.

CloudFront에서 AWS Config 설정

AWS Config를 설정할 때는 지원되는 모든 AWS 리소스를 기록하는 방법을 선택하거나, CloudFront의 변경 사항만 기록하는 등 특정 리소스의 구성 변경 사항만 기록하도록 지정할 수 있습니다. CloudFront에서 지원되는 특정 리소스는 AWS Config Developer Guide의 지원되는 AWS 리소스 유형을 참조하십시오.

CloudFront 배포에 대한 구성 변경 사항을 추적하려면 미국 동부(버지니아 북부) 퍼블릭 리전의 AWS 콘솔에 로그인해야 합니다.

Note

AWS Config를 사용하여 리소스를 기록할 때는 지연이 발생할 수 있습니다. AWS Config가 리소스를 발견한 후에만 리소스를 기록하기 때문입니다.

AWS Management 콘솔을 사용하여 CloudFront에서 AWS Config 설정

1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/config/에서 AWS Config 콘솔을 엽니다.
2. [Get Started Now]를 선택합니다.
3. 설정 페이지의 기록할 리소스 유형에서 AWS Config에 기록할 AWS 리소스 유형을 지정합니다. CloudFront 변경 사항만 기록하고 싶은 경우에는 특정 유형을 선택한 다음, CloudFront에서 변경 사항을 추적하려는 배포 또는 스트리밍 배포를 선택합니다.

   추적할 배포를 추가 또는 변경하려면 초기 설정을 완료한 후에 왼쪽의 설정을 선택합니다.
4. AWS Config에 필요한 추가 옵션(예: 알림 설정, 구성 정보의 위치 지정, 리소스 유형 평가를 위한 규칙 추가 등)을 지정합니다.

자세한 내용은 AWS Config Developer Guide의 콘솔을 이용한 AWS Config 설정을 참조하십시오.

AWS CLI 또는 API를 사용하여 CloudFront에서 AWS Config를 설정하는 방법은 다음 중 하나를 참조하십시오.

- AWS CLI 사용: AWS Config Developer Guide의 AWS CLI를 사용한 AWS Config 설정
- API 사용: AWS Config API Reference의 StartConfigurationRecorder 작업 및 기타 정보

CloudFront 구성 기록 보기

AWS Config가 배포에 구성 변경 사항을 기록하기 시작한 후에는 CloudFront에서 구성한 모든 배포의 구성 기록을 가져올 수 있습니다.

다음 방법 중 하나를 사용하여 구성 내역을 볼 수 있습니다:

- AWS Config 콘솔을 사용합니다. 기록된 리소스 각각에 대해 구성 세부 정보의 내역을 제공하는 타임라인 페이지를 볼 수 있습니다. 이 페이지를 보려면 전용 호스트 페이지의 Config 타임라인 열에서 회색 아이콘을 선택합니다. 자세한 내용은 AWS Config Developer Guide의 AWS Config 콘솔의 구성 세부 정보 보기를 참조하십시오.
- AWS CLI 명령을 실행합니다. 모든 배포 목록을 얻으려면 list-discovered-resources 명령을 사용합니다. 특정 기간 동안의 배포에 대한 구성 세부 정보를 가져오려면 get-resource-config-history 명령을 사용합니다. 자세한 내용은 AWS Config Developer Guide의 CLI를 사용한 구성 세부 정보 보기를 참조하십시오.
- 애플리케이션에서 AWS Config API를 사용합니다. 모든 배포 목록을 얻으려면 ListDiscoveredResources 작업을 사용합니다. 특정 기간 동안의 배포에 대한 구성 세부 정보를 가져오려면 GetResourceConfigHistory 작업을 사용합니다. 자세한 내용은 AWS Config API Reference를 참조하십시오.

예를 들어 AWS Config에서 모든 배포 목록을 가져오려면 다음과 같은 CLI 명령을 실행합니다.

```
aws configservice list-discovered-resources --resource-type
AWS::CloudFront::Distribution
```

또는 AWS Config에서 모든 RTMP 스트리밍 배포 목록을 가져오려면 다음과 같은 CLI 명령을 실행합니다.

```
aws configservice list-discovered-resources --resource-type
AWS::CloudFront::StreamingDistribution
```

액세스 로그 구성 및 사용

로그 파일을 생성하도록 CloudFront를 구성할 수 있습니다. 이러한 로그 파일에는 CloudFront에서 수신하는 모든 사용자 요청에 대한 세부 정보가 포함됩니다. 이러한 액세스 로그는 웹 및 RTMP 배포 모두에 사용할 수 있습니다. 로깅을 활성화하면 CloudFront에서 파일을 저장할 Amazon S3 버킷도 지정할 수 있습니다.

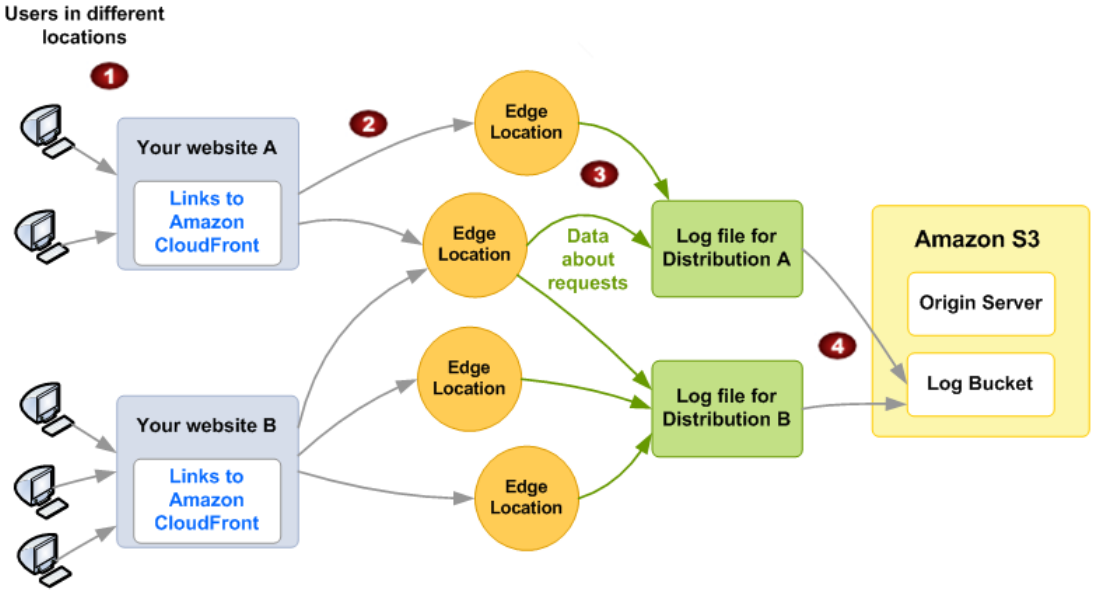
배포를 만들거나 업데이트할 때 로깅을 활성화할 수 있습니다. 자세한 내용은 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.

주제

- 로깅 작동 방식 (p. 394)
- 액세스 로그용 Amazon S3 버킷 선택 (p. 394)
- 로그 기록 및 로그 파일에 대한 액세스를 구성하는 데 필요한 권한 (p. 395)
- SSE-KMS 버킷에 사용할 경우 필요한 CMK 키 정책 (p. 396)
- 파일 이름 형식 (p. 396)
- 로그 파일 전송 시점 (p. 396)
- 요청 URL 또는 헤더가 최대 크기를 초과했을 때 요청이 기록되는 방법 (p. 397)
- 액세스 로그 분석 (p. 397)
- 로깅 설정 편집 (p. 397)
- Amazon S3 버킷에서 로그 파일 삭제 (p. 398)
- 로그 파일 형식 (p. 398)
- 액세스 로그 요금 (p. 410)

로깅 작동 방식

다음 다이어그램은 CloudFront에서 객체 요청에 대한 정보를 어떻게 기록하는지 보여 줍니다.



이전 다이어그램에 설명된 객체 요청에 대한 CloudFront 로그 정보가 아래에 나와 있습니다.

1. 이 다이어그램에서는 A와 B라는 두 웹 사이트가 있고 그에 해당하는 CloudFront 배포 두 개가 있습니다. 사용자가 배포와 연결된 URL을 사용하여 객체를 요청합니다.
2. CloudFront에서 각 요청을 해당 엣지 로케이션으로 라우팅합니다.
3. CloudFront에서 각 요청에 대한 데이터를 해당 배포 관련 로그 파일에 기록합니다. 이 예에서는 배포 A와 관련된 요청에 대한 정보가 배포 A 로그 파일에만 기록되고, 배포 B와 관련된 요청에 대한 정보가 배포 B 로그 파일에만 기록됩니다.
4. 로깅을 활성화할 당시에 지정한 Amazon S3 버킷에 배포하기 위해 CloudFront에서는 로그 파일을 정기적으로 저장합니다. 그런 다음 CloudFront에서는 배포를 위해 후속 요청에 대한 정보를 새 로그 파일에 저장하기 시작합니다.

지정된 시간 동안 콘텐츠에 액세스하는 사용자가 없는 경우 해당 시간 동안 로그 파일이 수신되지 않습니다.

로그 파일의 각 항목은 단일 요청에 대한 세부 정보를 제공합니다. 로그 파일 형식에 대한 자세한 내용은 로그 파일 형식 (p. 398) 단원을 참조하십시오.

> **Important**
>
> 모든 요청을 완전히 살펴보기 보다는 콘텐츠에 대한 요청 특성을 이해하는 데 로그를 사용하는 것이 좋습니다. CloudFront에서는 최대 효과에 기초하여 액세스 로그를 전송합니다. 요청에 따라서는 실제로 요청이 처리된 지 한참 후에 로그 레코드가 전송되거나 아예 전송되지 않을 수도 있습니다. 로그 항목이 액세스 로그에서 생략되는 경우 액세스 로그의 항목 수가 AWS 사용 보고서 및 결제 보고서에 표시되는 사용량과 일치하지 않습니다.

액세스 로그용 Amazon S3 버킷 선택

배포에 대해 로깅을 활성화하는 경우 CloudFront에서 로그 파일을 저장할 Amazon S3 버킷을 지정할 수 있습니다. Amazon S3을 오리진으로 사용할 경우에는 로그 파일에 동일한 버킷을 사용하지 않는 것이 좋습니다. 별도의 버킷을 사용하면 유지 관리가 간소화됩니다.

Note

아시아 태평양(홍콩) 또는 중동(바레인) 리전에서 Amazon S3 버킷을 선택하지 마십시오. CloudFront는 해당 리전의 버킷에 액세스 로그를 전달하지 않습니다.

동일 버킷에 여러 배포에 대한 로그 파일을 저장할 수 있습니다. 로깅을 활성화할 경우 파일 이름에 접두사(선택 사항)를 지정하여 배포와 연결되는 로그 파일을 추적할 수 있습니다.

로그 기록 및 로그 파일에 대한 액세스를 구성하는 데 필요한 권한

로그 파일용으로 지정하는 버킷에 대해 AWS 계정에 다음 권한이 있어야 합니다.

- 버킷의 S3 ACL(액세스 제어 목록)에서 `FULL_CONTROL` 권한을 부여받아야 합니다. 버킷 소유자인 경우 기본적으로 계정에 이 권한이 있습니다. 이 권한이 없을 경우 버킷 소유자가 버킷에 대한 ACL을 업데이트해야 합니다.
- `s3:GetBucketAcl`
- `s3:PutBucketAcl`

다음을 참조하십시오.

버킷에 대한 ACL

배포를 생성하거나 업데이트하고 로깅을 활성화할 경우 CloudFront는 이러한 권한을 사용하여 버킷에 대한 ACL을 업데이트함으로써 `awslogsdelivery` 계정에 `FULL_CONTROL` 권한을 부여합니다. `awslogsdelivery` 계정이 로그 파일을 버킷에 기록합니다. ACL을 업데이트하는 데 필요한 권한이 계정에 없을 경우 배포를 생성하거나 업데이트할 수 없습니다.

프로그래밍 방식으로 버킷 생성 요청을 제출했는데 지정한 이름의 버킷이 이미 존재하는 경우, S3는 해당 버킷에 대한 권한을 기본값으로 재설정합니다. 액세스 로그를 S3 버킷에 저장하도록 CloudFront를 구성했는데 더 이상 해당 버킷에서 로그를 가져오지 않는 경우, 버킷에 대한 권한을 점검하여 CloudFront가 필요한 권한을 가지고 있는지 확인하십시오.

버킷에 대한 ACL 복원

`awslogsdelivery` 계정에 대한 권한을 제거할 경우 CloudFront는 S3 버킷에 로그를 저장할 수 없습니다. CloudFront에서 배포에 대한 로그를 다시 저장할 수 있게 하려면 다음 중 하나를 수행하여 ACL 권한을 복원합니다.

- CloudFront의 배포에 대한 로깅을 비활성화한 다음 이를 다시 활성화합니다. 자세한 내용은 배포를 만들거나 업데이트할 때 지정하는 값 (p. 35) 단원을 참조하십시오.
- Amazon S3 콘솔의 S3 버킷으로 이동하고 권한을 추가하여 `awslogsdelivery`에 대한 ACL 권한을 수동으로 추가합니다. `awslogsdelivery`에 대한 ACL을 추가하려면 계정에 대해 정식 ID를 입력해야 합니다. 의 경우 다음과 같습니다.

  ```
  c4c1ede66af53448b93c283ce9448c4ba468c9432aa01d700d3878632f77d2d0
  ```

  S3 버킷에 ACL을 추가하는 방법에 대한 자세한 내용은 Amazon Simple Storage Service 콘솔 사용 설명서의 ACL 버킷 권한을 설정하는 방법을 참조하십시오.

각 로그 파일의 ACL

버킷의 ACL 외에도 각 로그 파일마다 ACL이 있습니다. 버킷 소유자에게는 각 로그 파일에 대한 `FULL_CONTROL` 권한이 있으며, 배포 소유자(버킷 소유자와 다른 경우)에게는 아무런 권한이 없습니다. `awslogsdelivery` 계정에는 읽기 및 쓰기 권한이 있습니다.

로깅 비활성화

로깅을 비활성화하더라도 CloudFront가 버킷이나 로그 파일에 대한 ACL이 자동으로 삭제되지 않습니다. 삭제하려면 사용자가 직접 삭제해야 합니다.

SSE-KMS 버킷에 사용할 경우 필요한 CMK 키 정책

고객 관리형 고객 마스터 키(CMK)와 함께 AWS KMS 관리형 키(SSE-KMS)를 사용하여 Amazon S3 버킷에 대해 서버 측 암호화를 활성화한 경우, CMK의 키 정책에 다음을 추가해야 버킷에 로그 파일을 쓸 수 있습니다. CloudFront는 로그 파일을 버킷에 업로드할 수 없기 때문에 기본 CMK를 사용할 수 없습니다.

```
{
    "Sid": "Allow CloudFront Flow Logs to use the key",
    "Effect": "Allow",
    "Principal": {
        "Service": "delivery.logs.amazonaws.com"
    },
    "Action": "kms:GenerateDataKey*",
    "Resource": "*"
}
```

파일 이름 형식

CloudFront에서 Amazon S3버킷에 저장하는 각 로그 파일의 이름에는 다음 파일 이름 형식이 사용됩니다.

<optional prefix>/*<distribution ID>*.*YYYY*-*MM*-*DD*-*HH*.*unique-ID*.gz

날짜 및 시간이 협정 세계시(UTC)로 표시됩니다.

예를 들어 접두사로 `example-prefix`를 사용하고 배포 ID가 `EMLARXS9EXAMPLE`이면 파일 이름은 다음과 같습니다.

```
example-prefix/EMLARXS9EXAMPLE.2019-11-14-20.RT4KCN4SGK9.gz
```

배포에 대한 로깅을 활성화할 경우 파일 이름에 접두사(선택 사항)를 지정하여 배포와 연결되는 로그 파일을 추적할 수 있습니다. 로그 파일 접두사의 값을 포함하고 접두사가 슬래시(/)로 끝나지 않으면 CloudFront에서 슬래시를 자동으로 추가합니다. 접두사가 슬래시로 끝나면 CloudFront에서 슬래시를 추가하지 않습니다.

파일 이름의 끝에 붙은 `.gz`는 CloudFront에서 로그 파일이 gzip으로 압축되었음을 나타냅니다.

로그 파일 전송 시점

CloudFront는 1시간 동안 여러 번 배포에 대한 액세스 로그를 전송합니다. 일반적으로, 로그 파일에는 지정된 기간에 CloudFront에서 수신한 요청에 대한 정보가 들어 있습니다. 보통, CloudFront에서는 로그에 이벤트가 기록된 후 한 시간 이내에 Amazon S3 버킷에 해당 기간에 대한 로그 파일을 전달합니다. 하지만 한 기간에 대한 일부 또는 전체 로그 파일 항목이 때때로 최대 24시간까지 지연되기도 합니다. 로그 항목이 지연되는 경우 CloudFront는 파일 이름에 파일이 전송된 날짜 및 시간이 아니라 요청이 발생한 기간의 날짜 및 시간이 로그 파일에 이러한 로그 항목을 저장합니다.

CloudFront는 로그 파일을 생성할 때 로그 파일에 포함되는 기간 중에 객체에 대한 요청을 받은 모든 엣지 로케이션의 배포 관련 정보를 통합합니다.

CloudFront는 CloudFront에서 배포와 연결된 객체에 대해 받는 요청 수에 따라 한 기간 동안 두 개 이상의 파일을 저장할 수 있습니다.

CloudFront는 로깅을 활성화하고 약 4시간이 지나면 액세스 로그들을 안정적으로 전송하기 시작합니다. 그 전에는 일부 액세스 로그들을 가져올 수 있습니다.

Note

해당 기간 동안 객체를 요청하는 사용자가 없으면 해당 기간 동안 아무런 로그 파일도 수신되지 않습니다.

요청 URL 또는 헤더가 최대 크기를 초과했을 때 요청이 기록되는 방법

쿠키를 포함하여 모든 요청 헤더의 총 크기가 20KB를 초과하거나 URL이 8192바이트를 초과하면 CloudFront가 요청을 완전히 구문 분석할 수 없고 요청을 기록할 수 없습니다. 요청이 기록되지 않으므로 로그 파일에 반환된 HTTP 오류 상태 코드가 표시되지 않습니다.

요청 본문이 최대 크기를 초과하면 HTTP 오류 상태 코드를 포함하여 요청이 기록됩니다.

액세스 로그 분석

시간당 액세스 로그를 여러 개 받을 수 있기 때문에, 지정된 기간에 받는 모든 로그 파일을 한 파일로 결합하는 것이 좋습니다. 그러고 나면 이 기간의 데이터를 보다 정확하고 완전하게 분석할 수 있습니다.

액세스 로그를 분석하기 위한 한 가지 방법으로 Amazon Athena를 사용할 수 있습니다. Athena는 CloudFront를 포함해 AWS 서비스용 데이터를 분석할 수 있는 대화형 쿼리 서비스입니다. 자세한 내용은 Amazon Athena 사용 설명서의 Amazon CloudFront 로그 쿼리를 참조하십시오.

또한 다음 AWS 블로그 게시물에서는 액세스 로그를 분석하는 몇 가지 방법을 설명합니다.

- Amazon CloudFront 요청 로깅(HTTP를 통해 제공되는 콘텐츠의 경우)
- Amazon CloudFront, 이제 스트리밍 액세스 로그 지원(RTMP를 통해 제공되는 콘텐츠의 경우)
- 개선된 CloudFront 로그, 이제 쿼리 문자열 사용

Important

모든 요청을 완전히 살펴보기 보다는 콘텐츠에 대한 요청 특성을 이해하는 데 로그를 사용하는 것이 좋습니다. CloudFront에서는 최대 효과에 기초하여 액세스 로그를 전송합니다. 요청에 따라서는 실제로 요청이 처리된 지 한참 후에 로그 레코드가 전송되거나 아예 전송되지 않을 수도 있습니다. 로그 항목이 액세스 로그에서 생략되는 경우 액세스 로그의 항목 수가 AWS 사용 보고서 및 결제 보고서에 표시되는 사용량과 일치하지 않습니다.

로깅 설정 편집

CloudFront 콘솔 또는 CloudFront API를 사용하여 로깅을 활성화/비활성화하거나, 로그가 저장되는 Amazon S3 버킷을 변경하고 로그 파일의 접두사를 변경할 수 있습니다. 로깅 설정에 대한 변경 내용은 12시간 내에 적용됩니다.

자세한 내용은 다음 주제를 참조하십시오.

- CloudFront 콘솔을 사용하여 웹 또는 RTMP 배포를 업데이트하려면 배포 업데이트 (p. 59) 단원을 참조하십시오.
- CloudFront API를 사용하여 웹 배포를 업데이트하려면 Amazon CloudFront API Reference의 UpdateDistribution을 참조하십시오.
- CloudFront API를 사용하여 RTMP 배포를 업데이트하려면 Amazon CloudFront API Reference의 UpdateStreamingDistribution을 참조하십시오.

CloudFront API를 사용하여 웹 배포의 액세스 로그 설정을 변경하려면 API 버전 `2009-04-02` 이상을 사용해야 합니다. CloudFront API를 사용하여 RTMP 배포의 액세스 로그 설정을 변경하려면 API 버전 `2010-05-01` 이상을 사용해야 합니다.

Amazon S3 버킷에서 로그 파일 삭제

CloudFront는 Amazon S3 버킷에서 로그 파일을 자동으로 삭제하지 않습니다. Amazon S3 버킷에서 로그 파일 삭제에 대한 자세한 내용은 다음 주제를 참조하십시오.

- Amazon S3 콘솔 사용: Amazon Simple Storage Service 콘솔 사용 설명서의 객체 삭제
- REST API 사용: Amazon Simple Storage Service API Reference의 DeleteObject

로그 파일 형식

주제

- 웹 배포 로그 파일 형식 (p. 399)
- RTMP 배포 로그 파일 형식 (p. 408)

로그 파일의 각 항목은 단일 최종 사용자 요청에 대한 세부 정보를 제공합니다. 웹 배포 로그 파일과 RTMP 배포 로그 파일이 동일하지는 않지만, 특성은 같습니다.

- W3C 확장 로그 파일 형식을 사용합니다.
- 탭으로 구분된 값을 포함합니다.
- 레코드를 포함합니다. 이 때 레코드가 시간 순서대로 나열되어 있지 않을 수도 있습니다.
- 두 헤더 행(파일 형식 버전이 표시된 행과 각 레코드에 포함된 W3C 필드가 나열된 행)을 포함합니다.
- 필드 값에서 공백 및 다른 특정 문자에 상응하는 URL 인코딩 값을 포함합니다.

  상응하는 URL 인코딩 값은 다음과 같은 문자에 사용됩니다.
  - ASCII 문자 코드 0~32
  - ASCII 문자 코드 127 이상
  - 다음 표의 모든 문자

  URL 인코딩 표준은 RFC 1738에 정의되어 있습니다.

URL 인코딩 값	문자
%3C	<
%3E	>
%22	"
%23	#
%25	%
%7B	{
%7D	}
%7C	|
%5C	\
%5E	^
%7E	~

URL 인코딩 값	문자
%5B	[
%5D	]
%60	`
%27	'
%20	공백

웹 배포 로그 파일 형식

웹 배포 로그 파일에는 다음 필드가 나열된 순서대로 포함됩니다.

필드 번호	필드 이름	설명
1	`date`	이벤트가 발생한 날짜가 `YYYY-MM-DD` 형식으로 표시됩니다. 예: `2019-06-30`. 날짜 및 시간이 협정 세계시(UTC)로 표시됩니다. WebSocket 연결의 경우, 연결이 종료된 날짜가 됩니다.
2	`time`	CloudFront 서버에서 요청 응답을 완료한 시간(UTC)입니다(예: `01:42:39`). WebSocket 연결의 경우, 연결이 종료된 시간이 됩니다.
3	`x-edge-location`	요청을 처리한 엣지 로케이션입니다. 각 엣지 로케이션은 3자 코드와 임의 배정된 번호로 식별됩니다(예: DFW3). 3자 코드는 일반적으로 엣지 로케이션 부근의 공항을 나타내는 국제 항공 수송 협회 공항 코드에 상응합니다. 이러한 약어는 향후에 변경될 수 있습니다. 엣지 로케이션 목록은 Amazon CloudFront 인프라 페이지를 참조하십시오.
4	`sc-bytes`	요청에 대한 응답으로 CloudFront가 최종 사용자에게 제공한 총 바이트 수(헤더 포함)입니다(예: `1045619`). WebSocket 연결의 경우, 연결을 통해 서버에서 클라이언트로 전송된 총 바이트 수입니다.
5	`c-ip`	요청을 한 최종 사용자의 IP 주소(예: `192.0.2.183` 또는 `2001:0db8:85a3:0000:0000:8a2e:0370:7334`)입니다. 최종 사용자가 HTTP 프록시 또는 로드 밸런서를 사용하여 요청을 전송하는 경우 `c-ip`의 값은 프록시 또는 로드 밸런서의 IP 주소 값입니다. 필드 20의 `X-Forwarded-For`도 참조하십시오.
6	`cs-method`	HTTP 요청 메서드: `DELETE`, `GET`, `HEAD`, `OPTIONS`, `PATCH`, `POST` 또는 `PUT`
7	`cs(Host)`	CloudFront 배포의 도메인 이름입니다(예: `d111111abcdef8.cloudfront.net`).
8	`cs-uri-stem`	경로와 객체를 식별하는 URI의 부분입니다(예: `/images/cat.jpg`). URL과 쿼리 문자열의 물음표(?)는 로그에 포함되지 않습니다.
9	`sc-status`	다음 값 중 하나입니다. • HTTP 상태 코드(예: `200`)입니다. HTTP 상태 코드 목록은 RFC 2616 섹션 10을 참조하십시오. 자세한 내용은 CloudFront에서 오리진의 HTTP 4xx 및 5xx 상태 코드를 처리 및 캐싱하는 방법 (p. 250) 단원을 참조하십시오. • `000` - CloudFront에서 요청에 응답하기 전에 브라우저 탭을 닫는 등의 동작으로 최종 사용자가 연결을 종료했음을 나타냅니다.

필드 번호	필드 이름	설명
		CloudFront가 응답을 전송하기 시작한 뒤에 최종 사용자가 연결을 종료하는 경우, 해당하는 HTTP 상태 코드가 로그에 포함됩니다.
10	`cs(Referer)`	요청을 시작한 도메인의 이름입니다. 일반적인 참조자에는 검색 엔진, 객체에 직접 연결하는 기타 웹 사이트 및 자체 웹 사이트가 포함됩니다.
11	`cs(User-Agent)`	요청의 `User-Agent` 헤더 값입니다. `User-Agent` 헤더는 요청을 제출한 디바이스의 유형 및 브라우저 등 요청의 소스를 식별하고 검색 엔진에서 요청이 온 경우에는 어느 검색 엔진인지 식별합니다. 자세한 내용은 User-Agent 헤더 (p. 244) 단원을 참조하십시오.
12	`cs-uri-query`	URI의 쿼리 문자열 부분(있는 경우)입니다. URI에 쿼리 문자열이 없는 경우 이 필드의 값은 하이픈(-)입니다. 자세한 내용은 쿼리 문자열 파라미터 기반의 콘텐츠 캐싱 (p. 196) 단원을 참조하십시오.
13	`cs(Cookie)`	이름-값 페어 및 관련 속성을 포함한 요청의 쿠키 헤더입니다. 쿠키 로깅을 활성화하는 경우 CloudFront는 오리진에 전달하도록 선택한 쿠키와 관계없이 모든 요청의 쿠키를 기록합니다. 요청에 쿠키 헤더가 포함되지 않은 경우 이 필드의 값은 하이픈(-)입니다. 쿠키에 대한 자세한 내용은 쿠키를 기반으로 콘텐츠 캐싱 (p. 198) 단원을 참조하십시오.

필드 번호	필드 이름	설명
14	`x-edge-result-type`	마지막 바이트가 엣지 로케이션을 떠난 후 CloudFront가 응답을 분류하는 방식입니다. 경우에 따라 결과 유형은 CloudFront가 응답을 전송할 준비가 된 시간과 CloudFront가 응답 전송을 완료한 시간 사이에 변경될 수 있습니다. 필드 23의 `x-edge-response-result-type`도 참조하십시오. 예를 들어, HTTP 스트리밍에서 CloudFront가 엣지 캐시에서 한 세그먼트를 찾는다고 가정해 보십시오. 이 시나리오에서 이 필드의 값은 보통 `Hit`입니다. 하지만 CloudFront가 전체 세그먼트, 즉 최종 결과 유형을 전달하기 전에 최종 사용자가 연결을 종료하면 이 필드의 값은 `Error`입니다. 또 다른 예로, 콘텐츠가 캐시 가능하지 않고 오리진 서버로 직접 다시 프록시되므로 WebSocket 연결에서 이 필드의 값이 `Miss`가 됩니다. 가능한 값에는 다음이 포함됩니다. • `Hit` – CloudFront가 엣지 캐시에서 최종 사용자에게 객체를 제공했습니다. 오리진의 응답에 `Cache-Control: no-cache` 헤더가 포함되어 있어도 CloudFront에서 결과 유형을 `Hit`로 분류하는 상황에 대한 정보는 동일 객체에 대한 동시 요청(트래픽 스파이크) (p. 244) 단원을 참조하십시오. • `RefreshHit` – CloudFront에서는 엣지 캐시에서 객체를 찾았지만 만료되었기 때문에 CloudFront가 오리진에 연결하여 캐시에 객체의 최신 버전이 포함되어 있는지 확인하고 포함되어 있지 않는 경우 최신 버전을 받을 것인지 결정합니다. • `Miss` – 엣지 캐시의 객체로는 요청을 충족할 수 없어서 CloudFront가 요청을 오리진 서버로 전달했으며 결과가 최종 사용자에게 반환되었습니다. • `LimitExceeded` – CloudFront 할당량(이전에는 제한이라고 함)이 초과되어 요청이 거부되었습니다. • `CapacityExceeded` – CloudFront 엣지 서버가 일시적으로 요청에 응답할 수 없어 CloudFront가 HTTP 503 상태 코드(서비스 사용 불가)를 반환했습니다. • `Error` – 일반적으로 이 오류는 요청의 결과로 클라이언트 오류(`sc-status`가 `4xx`) 또는 서버 오류(`sc-status`가 `5xx`)가 발생했음을 의미합니다. `sc-status`가 `200`이면 HTTP 요청에 성공했지만 모든 바이트를 다운로드하기 전에 클라이언트가 연결 해제된 것입니다. 이 필드의 값이 `Error`이고 `x-edge-response-result-type`의 값이 `Error`가 아니면 다운로드를 끝내기 전에 클라이언트 연결이 끊어진 것입니다. • `Redirect` – CloudFront가 HTTP에서 HTTPS로 리디렉션합니다. `sc-status`가 `403`이고 콘텐츠의 지리적 배포를 제한하도록 CloudFront를 구성한 경우 요청이 제한된 위치에서 들어올 수도 있습니다. 지리적 제한에 대한 자세한 내용은 콘텐츠의 지리적 배포 제한 (p. 183) 단원을 참조하십시오.
15	`x-edge-request-id`	요청을 고유하게 식별하는 암호화된 문자열입니다. 응답 헤더에서 이는 `x-amz-cf-id`입니다.

필드 번호	필드 이름	설명
16	`x-host-header`	최종 사용자가 이 요청에 대한 `Host` 헤더에 포함한 값입니다. 이 값은 요청의 도메인 이름입니다. • 객체 URL에 CloudFront 도메인 이름(예: http://d111111abcdef8.cloudfront.net/logo.png)을 사용하는 경우 이 필드에는 도메인 이름이 포함됩니다. • 객체 URL에 대체 도메인 이름(예: http://example.com/logo.png)을 사용하는 경우에는 이 필드에 대체 도메인 이름(예: example.com)이 포함됩니다. 대체 도메인 이름을 사용하려면 해당 이름을 배포에 추가해야 합니다. 자세한 내용은 대체 도메인 이름(CNAME)을 추가하여 파일에 대해 사용자 지정 URL 사용 (p. 66) 단원을 참조하십시오. 대체 도메인 이름을 사용하는 경우 배포와 연결된 도메인 이름은 `cs(Host)` 필드7을 참조하십시오.
17	`cs-protocol`	최종 사용자가 요청에 지정한 프로토콜은 `http`, `https`, `ws` 또는 `wss`입니다.
18	`cs-bytes`	최종 사용자가 요청에 포함한 데이터의 바이트 수(헤더 포함)입니다. WebSocket 연결의 경우, 연결에서 클라이언트로부터 서버로 전송된 총 바이트 수입니다.
19	`time-taken`	CloudFront 엣지 서버가 최종 사용자의 요청을 받는 시점과 CloudFront가 엣지 서버의 출력 대기열에 응답의 최종 바이트를 쓰는 시점 간의 시간 차이(1/1000초 단위까지 측정, 예: 0.002)로, 서버에서 측정됩니다. 최종 사용자 관점에서 보면 전체 객체를 가져오는 데 걸리는 총 시간은 네트워크 지연 시간과 TCP 버퍼링으로 인해 이 값보다 큽니다.
20	`x-forwarded-for`	최종 사용자가 HTTP 프록시 또는 로드 밸런서를 사용하여 요청을 전송하는 경우 필드 5의 `c-ip`의 값은 프록시 또는 로드 밸런서의 IP 주소 값입니다. 이 경우 이 필드는 요청을 시작한 최종 사용자의 IP 주소입니다. 해당하는 경우 이 필드에는 IPv4 주소(예: 192.0.2.44)와 IPv6 주소(예: 2001:0db8:85a3:0000:0000:8a2e:0370:7334)가 포함됩니다. 최종 사용자가 HTTP 프록시 또는 로드 밸런서를 사용하지 않은 경우 `x-forwarded-for`의 값은 하이픈(-)입니다.
21	`ssl-protocol`	필드 17의 `cs-protocol`이 `https`인 경우 이 필드에는 클라이언트와 CloudFront가 요청과 응답 전송을 협상한 SSL/TLS 프로토콜이 포함됩니다. 가능한 값은 다음을 포함합니다. • `SSLv3` • `TLSv1` • `TLSv1.1` • `TLSv1.2` 필드 17의 `cs-protocol`이 `http`인 경우 이 필드의 값은 하이픈(-)입니다.

필드 번호	필드 이름	설명
22	`ssl-cipher`	필드 17의 `cs-protocol`이 `https`인 경우 이 필드에는 클라이언트와 CloudFront가 요청과 응답 암호화를 협상한 SSL/TLS 암호가 포함됩니다. 가능한 값은 다음을 포함합니다. • `ECDHE-RSA-AES128-GCM-SHA256` • `ECDHE-RSA-AES128-SHA256` • `ECDHE-RSA-AES128-SHA` • `ECDHE-RSA-AES256-GCM-SHA384` • `ECDHE-RSA-AES256-SHA384` • `ECDHE-RSA-AES256-SHA` • `AES128-GCM-SHA256` • `AES256-GCM-SHA384` • `AES128-SHA256` • `AES256-SHA` • `AES128-SHA` • `DES-CBC3-SHA` • `RC4-MD5` 필드 17의 `cs-protocol`이 `http`인 경우 이 필드의 값은 하이픈(-)입니다.

필드 번호	필드 이름	설명
23	`x-edge-response-result-type`	CloudFront가 최종 사용자에게 응답을 반환하기 직전에 응답을 분류한 방식입니다. 필드 14의 `x-edge-result-type`도 참조하십시오. 가능한 값에는 다음이 포함됩니다. • `Hit` – CloudFront가 엣지 캐시에서 최종 사용자에게 객체를 제공했습니다. • `RefreshHit` – CloudFront가 엣지 캐시의 객체를 찾았지만 객체가 만료되어 CloudFront가 오리진에 연결해서 캐시에 최신 객체 버전이 있는지 확인했습니다. • `Miss` – 엣지 캐시의 객체로는 요청을 충족할 수 없어서 CloudFront가 요청을 오리진 서버로 전달했으며 결과가 최종 사용자에게 반환되었습니다. • `LimitExceeded` – CloudFront 할당량(이전에는 제한이라고 함)이 초과되어 요청이 거부되었습니다. • `CapacityExceeded` – 객체 제공을 요청하는 시점에 엣지 로케이션에 용량이 충분하지 않아 CloudFront에서 `503` 오류가 반환되었습니다. • `Error` – 일반적으로 이 오류는 요청의 결과로 클라이언트 오류(`sc-status`가 `4xx`) 또는 서버 오류(`sc-status`가 `5xx`)가 발생했음을 의미합니다. `x-edge-result-type`의 값이 `Error`이고 이 필드의 값이 `Error`가 아닌 경우 다운로드를 완료하기 전에 클라이언트 연결이 끊어졌습니다. • `Redirect` – CloudFront가 HTTP에서 HTTPS로 리디렉션합니다. `sc-status`가 `403`이고 콘텐츠의 지리적 배포를 제한하도록 CloudFront를 구성한 경우 요청이 제한된 위치에서 들어올 수도 있습니다. 지리적 제한에 대한 자세한 내용은 콘텐츠의 지리적 배포 제한 (p. 183) 단원을 참조하십시오.
24	`cs-protocol-version`	최종 사용자가 요청에서 지정한 HTTP 버전입니다. 가능한 값은 다음과 같습니다. • `HTTP/0.9` • `HTTP/1.0` • `HTTP/1.1` • `HTTP/2.0`

필드 번호	필드 이름	설명
25	`fle-status`	배포에 대해 필드 레벨 암호화가 구성될 때, 요청 본문이 성공적으로 처리되었는지를 나타내는 코드가 이 필드에 포함됩니다. 배포에 대해 필드 레벨 암호화가 구성되지 않은 경우 이 필드의 값은 하이픈(-)입니다. CloudFront가 성공적으로 요청 본문을 처리하고, 지정된 필드의 값을 암호화하고, 요청을 오리진으로 전달할 때 이 필드의 값은 `Processed`입니다. 이 경우 필드 14 `x-edge-result-type`의 값은 계속해서 클라이언트 측 또는 서버 측 오류를 나타낼 수 있습니다. 요청이 필드 레벨 암호화 할당량을 초과하는 경우 `fle-status`는 다음 오류 코드 중 하나를 포함하며, CloudFront는 HTTP 상태 코드 `400`을 최종 사용자에게 반환합니다. 필드 레벨 암호화의 현재 할당량에 대한 목록은 필드 레벨 암호화에 대한 할당량 (p. 450) 단원을 참조하십시오. • `FieldLengthLimitClientError` – 암호화되도록 구성된 필드가 허용되는 최대 길이를 초과했습니다. • `FieldNumberLimitClientError` – CloudFront가 암호화하도록 구성된 요청에 허용된 필드의 수보다 많은 필드가 있습니다. • `RequestLengthLimitClientError` – 필드 레벨 암호화 구성 시 요청 본문의 길이가 허용되는 최대 길이를 초과했습니다. 이 필드에 사용할 수 있는 다른 값에는 다음이 포함됩니다. • `ForwardedByContentType` – 어떠한 콘텐츠도 구성되지 않았기 때문에 CloudFront 가 요청을 구문 분석 또는 암호화 없이 오리진에 전달했습니다. • `ForwardedByQueryArgs` – 요청에 필드 레벨 암호화 구성에 없는 쿼리 인수를 포함되기 때문에 CloudFront가 요청을 구문 분석 또는 암호화 없이 오리진에 전달했습니다. • `ForwardedDueToNoProfile` – 필드 레벨 암호화 구성에 어떠한 프로필도 지정되지 않았기 때문에 CloudFront가 요청을 구문 분석 또는 암호화 없이 오리진에 전달했습니다. • `MalformedContentTypeClientError` – `Content-Type` 헤더의 값이 유효하지 않은 형식이기 때문에 CloudFront가 요청을 거부하고 최종 사용자에게 HTTP `400` 상태 코드를 반환했습니다. • `MalformedInputClientError` – 요청 본문이 유효하지 않은 형식이기 때문에 CloudFront가 요청을 거부하고 최종 사용자에게 HTTP `400` 상태 코드를 반환했습니다. • `MalformedQueryArgsClientError` – 쿼리 인수가 비어 있거나 유효하지 않은 형식이기 때문에 CloudFront가 요청을 거부하고 최종 사용자에게 HTTP `400` 상태 코드를 반환했습니다. • `RejectedByContentType` – 필드 레벨 암호화 구성에 어떠한 콘텐츠 유형도 지정되지 않았기 때문에 CloudFront가 요청을 거부하고 최종 사용자에게 HTTP `400` 상태 코드를 반환했습니다. • `RejectedByQueryArgs` – 필드 레벨 암호화 구성에 어떠한 쿼리 인수도 지정되지 않았기 때문에 CloudFront가 요청을 거부하고 최종 사용자에게 HTTP `400` 상태 코드를 반환했습니다. • `ServerError` – 서버가 오류를 반환했습니다.

필드 번호	필드 이름	설명
26	`fle-encrypted-fields`	CloudFront가 암호화하고 오리진에 전달하는 필드의 수입니다. CloudFront는 데이터를 암호화하면서 처리된 요청을 오리진으로 스트리밍합니다. 따라서 `fle-status`의 값이 error인 경우에도 `fle-encrypted-fields`는 값을 가질 수 있습니다. 배포에 대해 필드 레벨 암호화가 구성되지 않은 경우 `fle-encrypted-fields`의 값은 하이픈(`-`)입니다.
27	`c-port`	최종 사용자가 보낸 요청의 포트 번호입니다.
28	`time-to-first-byte`	요청을 수신한 후 응답의 첫 바이트를 쓸 때까지의 시간(초)이며 서버에서 측정합니다.

필드 번호	필드 이름	설명
29	`x-edge-detailed-result-type`	`x-edge-result-type`(필드 14)이 `Error`가 아닌 경우 이 필드는 `x-edge-result-type`과 동일한 값을 포함합니다. `x-edge-result-type`이 `Error`인 경우 이 필드는 특정 유형의 오류를 포함합니다. 이 필드에 사용할 수 있는 오류 유형 값에는 다음이 포함됩니다. • `AbortedOrigin` — CloudFront에 오리진 관련 문제가 발생했습니다. • `ClientCommError` — CloudFront와 최종 사용자 간의 통신 문제로 인해 최종 사용자에 대한 응답이 중단되었습니다. • `ClientGeoBlocked` — 최종 사용자의 지리적 위치에서 요청을 거부하도록 배포가 구성됩니다. • `ClientHungUpRequest` — 요청을 보내는 동안 최종 사용자가 중간에 중지했습니다. • `Error` – 오류 유형이 기타 모든 카테고리에 적합하지 않아 오류가 발생했습니다. 이러한 오류 유형은 CloudFront가 CloudFront 캐시에서 오류 응답을 제공할 때 발생할 수 있습니다. • `InvalidRequest` — CloudFront가 최종 사용자로부터 잘못된 요청을 받았습니다. • `InvalidRequestBlocked` – 요청한 리소스에 대한 액세스가 차단됩니다. • `InvalidRequestCertificate` — 배포가 HTTPS 연결이 설정된 SSL/TLS 인증서와 일치하지 않습니다. • `InvalidRequestHeader` — 요청에 잘못된 헤더 필드 또는 값이 포함되어 있습니다. • `InvalidRequestMethod` — 사용되었던 HTTP 요청 메서드를 처리하도록 배포가 구성되지 않았습니다. 이러한 문제는 배포에서 캐시 가능한 요청만 지원하는 경우 발생할 수 있습니다. • `OriginConnectError` — CloudFront가 오리진에 연결할 수 없습니다. • `OriginContentRangeLengthError` — 오리진 응답의 `Content-Length` 헤더가 `Content-Range` 헤더의 길이와 일치하지 않습니다. • `OriginDnsError` - CloudFront가 오리진의 도메인 이름을 확인할 수 없습니다. • `OriginError` — 오리진에서 잘못된 응답을 반환했습니다. • `OriginHeaderTooBigError` — 오리진에서 반환한 헤더가 너무 커서 CloudFront가 처리할 수 없습니다. • `OriginInvalidResponseError` — 오리진에서 잘못된 응답을 반환했습니다. • `OriginReadError` - CloudFront가 오리진에서 읽을 수 없습니다. • `OriginWriteError` - CloudFront가 오리진에 쓸 수 없습니다. • `OriginZeroSizeObjectError` — 오리진에서 전송된 크기가 0인 객체로 인해 오류가 발생했습니다. • `SlowReaderOriginError` — 최종 사용자가 오리진 오류를 일으킨 메시지를 느리게 읽었습니다.
30	`sc-content-type`	응답의 HTTP `Content-Type` 헤더 값입니다.

필드 번호	필드 이름	설명
31	`sc-content-len`	응답의 HTTP `Content-Length` 헤더 값입니다.
32	`sc-range-start`	응답에 HTTP `Content-Range` 헤더가 포함되어 있으면 이 필드에 범위 시작 값이 포함됩니다.
33	`sc-range-end`	응답에 HTTP `Content-Range` 헤더가 포함되어 있으면 이 필드에 범위 끝 값이 포함됩니다.

다음은 웹 배포 로그 파일 예입니다.

```
#Version: 1.0
#Fields: date time x-edge-location sc-bytes c-ip cs-method cs(Host) cs-uri-stem sc-status
 cs(Referer) cs(User-Agent) cs-uri-query cs(Cookie) x-edge-result-type x-edge-request-id
 x-host-header cs-protocol cs-bytes time-taken x-forwarded-for ssl-protocol ssl-cipher x-
edge-response-result-type cs-protocol-version fle-status fle-encrypted-fields c-port time-
to-first-byte x-edge-detailed-result-type sc-content-type sc-content-len sc-range-start sc-
range-end
2019-12-04 21:02:31 LAX1 392 192.0.2.100 GET d111111abcdef8.cloudfront.net /index.html
 200 - Mozilla/5.0%20(Windows%20NT%2010.0;%20Win64;%20x64)%20AppleWebKit/537.36%20(KHTML,
%20like%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Hit
 SOX4xwn4XV6Q4rgb7XiVGOHms_BGlTAC4KyHmureZmBNrjGdRLiNIQ== d111111abcdef8.cloudfront.net
 https 23 0.001 - TLSv1.2 ECDHE-RSA-AES128-GCM-SHA256 Hit HTTP/2.0 - - 11040 0.001 Hit
 text/html 78 - -
2019-12-04 21:02:31 LAX1 392 192.0.2.100 GET d111111abcdef8.cloudfront.net /index.html
 200 - Mozilla/5.0%20(Windows%20NT%2010.0;%20Win64;%20x64)%20AppleWebKit/537.36%20(KHTML,
%20like%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Hit
 k6WGMNkEzR5BEM_SaF47gjtX9zBDO2m349OY2an0QPEaUum1ZOLrow== d111111abcdef8.cloudfront.net
 https 23 0.000 - TLSv1.2 ECDHE-RSA-AES128-GCM-SHA256 Hit HTTP/2.0 - - 11040 0.000 Hit
 text/html 78 - -
2019-12-04 21:02:31 LAX1 392 192.0.2.100 GET d111111abcdef8.cloudfront.net /index.html
 200 - Mozilla/5.0%20(Windows%20NT%2010.0;%20Win64;%20x64)%20AppleWebKit/537.36%20(KHTML,
%20like%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Hit
 f37nTMVvnKvV2ZSvEsivup_c2kZ7VXzYdjC-GUQZ5qNs-89BlWazbw== d111111abcdef8.cloudfront.net
 https 23 0.001 - TLSv1.2 ECDHE-RSA-AES128-GCM-SHA256 Hit HTTP/2.0 - - 11040 0.001 Hit
 text/html 78 - -
2019-12-13 22:36:27 SEA19-C1 900 192.0.2.200 GET d111111abcdef8.cloudfront.net /
favicon.ico 502 http://www.example.com/ Mozilla/5.0%20(Windows
%20NT%2010.0;%20Win64;%20x64)%20AppleWebKit/537.36%20(KHTML,
%20like%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Error
 1pkpNfBQ39sYMnjjUQjmH2w1wdJnbHYTbag21o_3OfcQgPzdL2RSSQ== www.example.com http 675 0.102 -
 - - Error HTTP/1.1 - - 25260 0.102 OriginDnsError text/html 507 - -
2019-12-13 22:36:26 SEA19-C1 900 192.0.2.200 GET d111111abcdef8.cloudfront.net / 502
 - Mozilla/5.0%20(Windows%20NT%2010.0;%20Win64;%20x64)%20AppleWebKit/537.36%20(KHTML,
%20like%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Error
 3AqrZGCnF_g0-5KOvfA7c9XLcf4YGvMFSeFdIetR1N_2y8jSis8Zxg== www.example.com http 735 0.107 -
 - - Error HTTP/1.1 - - 3802 0.107 OriginDnsError text/html 507 - -
2019-12-13 22:37:02 SEA19-C2 900 192.0.2.200 GET d111111abcdef8.cloudfront.net / 502
 - curl/7.55.1 - - Error kBkDzGnceVtWHqSCqBUqtA_cEs2T3tFUBbnBNkB9El_uVRhHgcZfcw==
 www.example.com http 387 0.103 - - - Error HTTP/1.1 - - 12644 0.103 OriginDnsError text/
html 507 - -
```

RTMP 배포 로그 파일 형식

RTMP 액세스 로그의 각 레코드에는 연결, 재생, 일시 중지, 중지, 연결 해제 등 재생 이벤트를 나타냅니다. 따라서 CloudFront는 최종 사용자가 비디오를 볼 때마다 여러 로그 레코드를 생성합니다. 동일 스트림 ID에서 발생되는 로그 레코드를 연결하려면 `x-sid` 필드를 사용합니다.

Note

모든 이벤트에 대한 값이 있는 필드도 있고 재생, 중지, 일시 중지, 일시 중지 취소 및 탐색 이벤트에 대한 값만 있는 필드도 있습니다. 일반적으로 로그 파일에서 필드에 대해 하이픈(-)이 포함되는 경우 이러한 필드는 해당 이벤트와 관련이 없다는 의미입니다.

다음 표에서는 이벤트 유형과 관계 없이 RTMP 배포 로그 파일의 각 레코드에 있는 필드를 설명합니다. 필드는 나열된 순서대로 로그에 표시됩니다.

필드 번호	필드 이름	설명
1	`date`	이벤트가 발생한 날짜가 YYYY-MM-DD 형식으로 표시됩니다(예: 2019-05-23). 날짜 및 시간이 협정 세계시(UTC)로 표시됩니다.
2	`time`	서버에서 요청을 받은 시간(in UTC)입니다. 예: 01:42:39.
3	`x-edge-location`	재생 이벤트가 발생한 엣지 로케이션입니다. 각 엣지 로케이션은 3자 코드와 임의 배정된 번호로 식별됩니다(예: DFW3). 3자 코드는 일반적으로 엣지 로케이션 부근의 공항을 나타내는 국제 항공 수송 협회 공항 코드에 상응합니다. 이러한 약어는 향후에 변경될 수 있습니다. 엣지 로케이션 목록은 Amazon CloudFront 인프라 페이지를 참조하십시오.
4	`c-ip`	클라이언트 IP(예: 192.0.2.183)입니다.
5	`x-event`	이벤트 유형입니다. 이 값은 [Connect], [Disconnect], [Play], [Stop], [Pause], [Unpause] 또는 [Seek] 이벤트입니다.
6	`sc-bytes`	이벤트 시점까지 서버에서 클라이언트로 보낸 총 바이트 누계입니다.
7	`x-cf-status`	이벤트 상태를 나타내는 코드입니다. 현재는 "OK"가 이 필드의 유일한 값입니다. 향후 새 기능이 추가되면 새 상태가 필요할 수도 있습니다.
8	`x-cf-client-id`	클라이언트를 구분하는 데 사용할 수 있는 불투명 문자열 식별자입니다. 이 값은 각 연결마다 고유합니다.
9	`cs-uri-stem`	애플리케이션과 애플리케이션 인스턴스를 포함하는 URL의 스템 부분입니다. 경우에 따라서는 FMS 연결 문자열이라고도 합니다. 예: `rtmp://shqshne4jdp4b6.cloudfront.net/cfx/st`. URL과 쿼리 문자열의 물음표(?)는 로그에 포함되지 않습니다.
10	`cs-uri-query`	연결 문자열에 포함되는 URL의 쿼리 문자열 부분입니다. URL과 쿼리 문자열의 물음표(?)는 로그에 포함되지 않습니다.
11	`c-referrer`	참조자의 URI입니다.
12	`x-page-url`	SWF가 연결된 페이지의 URL입니다.
13	`c-user-agent`	요청의 `User-Agent` 헤더 값입니다. `User-Agent` 헤더는 요청을 제출한 디바이스 유형을 식별합니다. 자세한 내용은 User-Agent 헤더 (p. 244) 단원을 참조하십시오.

다음 필드에는 일반적으로 재생, 중지, 일시 중지, 일시 중지 취소 및 탐색 이벤트에 대한 값만 있습니다. 다른 이벤트의 경우에는 단일 하이픈(-)이 포함됩니다. 이러한 필드는 로그에서 이전 표의 필드 뒤에, 나열된 순서대로 표시됩니다.

필드 번호	필드 이름	설명
14	`x-sname`	스트림 이름입니다.
15	`x-sname-query`	스트림 쿼리 문자열(있는 경우)입니다.
16	`x-file-ext`	스트림 유형(예: FLV)입니다.
17	`x-sid`	스트림 ID입니다. 이 값은 연결에 사용되는 고유 정수 식별자입니다.

다음은 RTMP 배포 로그 파일 예입니다.

```
#Version: 1.0
#Fields: date time x-edge-location c-ip x-event sc-bytes x-cf-status x-cf-client-id cs-uri-
stem cs-uri-query c-referrer x-page-url c-user-agent x-sname x-sname-query x-file-ext x-sid
2010-03-12  23:51:20  SEA4  192.0.2.147  connect  2014  OK
 bfd8a98bee0840d9b871b7f6ade9908f  rtmp://shqshne4jdp4b6.cloudfront.net/cfx/st key=value
  http://player.example.com/player.swf  http://www.example.com/support/jw-player-setup-
wizard?example=204  LNX%2010,0,32,18  -  -  -  -
2010-03-12  23:51:21  SEA4  192.0.2.222  play  3914  OK  bfd8a98bee0840d9b871b7f6ade9908f
  rtmp://shqshne4jdp4b6.cloudfront.net/cfx/st key=value  http://player.example.com/
player.swf  http://www.example.com/support/jw-player-setup-wizard?example=204  LNX
%2010,0,32,18  myvideo  p=2&q=4  flv  1
2010-03-12  23:53:44  SEA4  192.0.2.4  stop  323914  OK  bfd8a98bee0840d9b871b7f6ade9908f
  rtmp://shqshne4jdp4b6.cloudfront.net/cfx/st key=value  http://player.example.com/
player.swf  http://www.example.com/support/jw-player-setup-wizard?example=204  LNX
%2010,0,32,18  dir/other/myvideo  p=2&q=4  flv  1
2010-03-12  23:53:44  SEA4  192.0.2.103  play  8783724  OK
 bfd8a98bee0840d9b871b7f6ade9908f  rtmp://shqshne4jdp4b6.cloudfront.net/cfx/st key=value
  http://player.example.com/player.swf  http://www.example.com/support/jw-player-setup-
wizard?example=204  LNX%2010,0,32,18  dir/favs/myothervideo  p=42&q=14  mp4  2
2010-03-12  23:56:21  SEA4  192.0.2.199  stop  429822014  OK
 bfd8a98bee0840d9b871b7f6ade9908f  rtmp://shqshne4jdp4b6.cloudfront.net/cfx/st key=value
  http://player.example.com/player.swf  http://www.example.com/support/jw-player-setup-
wizard?example=204  LNX%2010,0,32,18  dir/favs/myothervideo  p=42&q=14  mp4  2
2010-03-12  23:59:44  SEA4  192.0.2.14  disconnect  429824092  OK
 bfd8a98bee0840d9b871b7f6ade9908f  rtmp://shqshne4jdp4b6.cloudfront.net/cfx/st key=value
  http://player.example.com/player.swf  http://www.example.com/support/jw-player-setup-
wizard?example=204  LNX%2010,0,32,18  -  -  -  -
```

액세스 로그 요금

액세스 로깅은 CloudFront의 선택적 기능입니다. 액세스 로깅을 활성화하더라도 별도 요금이 부과되지 않습니다. 하지만 Amazon S3에 파일을 저장하고 액세스하는 데는 일반적인 Amazon S3 요금이 부과되며, 사용자는 언제라도 이러한 파일을 삭제할 수 있습니다.

Amazon S3 요금에 대한 자세한 내용은 Amazon S3 요금을 참조하십시오.

CloudFront 요금에 대한 자세한 내용은 CloudFront 요금을 참조하십시오.

AWS CloudTrail를 사용하여 CloudFront API로 전송된 요청 캡처

IAM 사용자를 포함하여 AWS 계정에 의해 CloudFront API로 전송되는 모든 요청에 대한 정보를 캡처하는 AWS 서비스인 CloudTrail이 CloudFront와 통합되었습니다. CloudTrail은 이러한 요청의 로그 파일을 지정

된 Amazon S3 버킷에 정기적으로 저장합니다. CloudTrail은 CloudFront 콘솔, CloudFront API, AWS SDK, CloudFront CLI 또는 또 다른 서비스(예: AWS CloudFormation) 등 요청이 생성되는 데 사용된 것이 무엇이든 상관 없이 모든 요청에 대한 정보를 캡처합니다.

CloudTrail 로그 파일의 정보를 사용하여 CloudFront에 대해 생성된 요청, 각 요청이 생성된 소스 IP 주소, 요청을 생성한 사람, 요청 생성 시기 등을 확인할 수 있습니다. 그 구성 및 활성화 방법을 포함하여 CloudTrail에 대한 자세한 내용은 AWS CloudTrail User Guide를 참조하십시오.

> **Note**
>
> CloudFront는 전역적 서비스입니다. CloudTrail 로그에서 CloudFront 요청을 보려면 기존 추적을 업데이트하여 전역적 서비스를 포함시켜야 합니다. 자세한 내용은 AWS CloudTrail User Guide의 Updating a Trail 및 About Global Service Events 단원을 참조하십시오.

주제

- CloudTrail의 CloudFront 정보 (p. 411)
- CloudFront 로그 파일 항목 이해 (p. 412)

CloudTrail의 CloudFront 정보

CloudTrail은 계정 생성 시 AWS 계정에서 활성화됩니다. CloudFront에서 활동이 수행되면 해당 활동은 이벤트 기록에서 다른 AWS 서비스 이벤트와 함께 CloudTrail 이벤트에 기록됩니다. AWS 계정에서 최신 이벤트를 확인, 검색 및 다운로드할 수 있습니다. CloudFront는 전 세계 서비스이므로 이 서비스에 대한 이벤트는 미국 동부(버지니아 북부)에 기록됩니다. 자세한 내용은 CloudTrail 이벤트 기록에서 이벤트 보기를 참조하십시오.

CloudFront 이벤트를 포함하여 AWS 계정에 이벤트를 지속적으로 기록하려는 경우 추적을 생성합니다. 이 추적은 전 세계 서비스 이벤트를 포함해야 합니다. 추적은 CloudTrail이 Amazon S3 버킷으로 로그 파일을 전송할 수 있도록 합니다. 기본적으로 콘솔에서 추적을 생성할 때 모든 리전에 추적이 적용되며 전 세계 서비스 이벤트가 포함됩니다. 추적은 AWS 파티션에 있는 모든 리전의 이벤트를 로깅하고 지정된 Amazon S3 버킷으로 로그 파일을 전송합니다. 또는 CloudTrail 로그에서 수집된 이벤트 데이터를 추가 분석 및 처리하도록 다른 AWS 서비스를 구성할 수 있습니다. 자세한 내용은 다음을 참조하십시오.

- 추적 생성 개요
- CloudTrail 지원 서비스 및 통합
- CloudTrail에 대한 Amazon SNS 알림 구성
- 여러 리전에서 CloudTrail 로그 파일 받기 및 여러 계정에서 CloudTrail 로그 파일 받기

모든 CloudFront API 작업이 CloudTrail에서 로깅되고 Amazon CloudFront API Reference에 문서화됩니다. 예를 들어 `CreateDistribution`, `GetDistribution` 및 `ListInvalidations` API에 대한 호출은 CloudTrail 로그 파일에 항목을 생성합니다.

모든 이벤트 및 로그 항목에는 요청을 생성한 사용자에 대한 정보가 들어 있습니다. 자격 증명 정보를 이용하면 다음을 쉽게 판단할 수 있습니다.

- 요청을 루트로 했는지 아니면 IAM 사용자 자격 증명으로 했는지 여부
- 역할 또는 연합된 사용자에 대한 임시 보안 자격 증명을 사용하여 요청이 생성되었는지 여부.
- 다른 AWS 서비스에서 요청했는지 여부.

자세한 내용은 CloudTrail userIdentity 요소를 참조하십시오.

CloudFront 로그 파일 항목 이해

각 JSON 형식의 CloudTrail 로그 파일에는 하나 이상의 로그 항목이 포함될 수 있습니다. 로그 항목은 어떤 소스로부터의 단일 요청을 나타내며 모든 매개 변수, 작업 날짜와 시간 등 요청한 작업에 대한 정보가 포함됩니다. 로그 항목은 특정 순서를 유지하지 않으며, API 호출의 정렬된 스택 추적 정보가 아닙니다.

`eventName` 요소는 발생한 작업과 해당 작업을 수행하는 데 사용된 API 버전을 식별합니다. 예를 들면 다음의 `eventName` 값은 웹 배포가 업데이트되었으며 작업을 수행하는 데 2014-01-31 API 버전이 사용되었음을 나타냅니다.

```
UpdateDistribution2014_01_31
```

다음은 5개 작업을 보여 주는 CloudTrail 로그 항목이 나타낸 예제입니다.

- 웹 배포 구성을 업데이트합니다. `eventName`의 값이 `UpdateDistribution`입니다.
- 현재 계정과 연결된 웹 배포를 나열합니다. `eventName`의 값이 `ListDistributions`입니다.
- 특정 웹 배포에 대한 구성을 가져옵니다. `eventName`의 값이 `GetDistribution`입니다.
- 무효화 배치 요청을 생성합니다. `eventName`의 값이 `CreateInvalidation`입니다.
- 현재 계정과 연결된 원본 액세스 ID를 나열합니다. `eventName`의 값이 `ListCloudFrontOriginAccessIdentities`입니다.


```
{
   "Records": [{
      "eventVersion": "1.01",
      "userIdentity": {
         "type": "IAMUser",
         "principalId": "A1B2C3D4E5F6G7EXAMPLE",
         "arn": "arn:aws:iam::111122223333:user/smithj",
         "accountId": "111122223333",
         "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
         "userName": "smithj"
      },
      "eventTime": "2014-05-06T18:00:32Z",
      "eventName": "UpdateDistribution2014_01_31",
      "sourceIPAddress": "192.0.2.17",
      "userAgent": "aws-sdk-ruby/1.39.0 ruby/1.9.3 x86_64-linux",
      "requestParameters": {
         "id": "EDFDVBD6EXAMPLE",
         "ifMatch": "E9LHASXEXAMPLE",
         "distributionConfig": {
            "restrictions": {
               "geoRestriction": {
                  "quantity": 0,
                  "restrictionType": "none"
               }
            },
            "customErrorResponses": {
               "quantity": 0
            },
            "defaultRootObject": "index.html",
            "aliases": {
               "quantity": 1,
               "items": ["example.com"]
            },
            "logging": {
               "bucket": "",
               "enabled": false,
               "prefix": "",
               "includeCookies": false
```

```
            },
            "viewerCertificate": {
               "iAMCertificateId": "A1B2C3D4E5F6G7EXAMPLE",
               "sSLSupportMethod": "sni-only"
            },
            "callerReference": "2014-05-06 64832",
            "defaultCacheBehavior": {
               "targetOriginId": "Images",
               "allowedMethods": {
                  "items": ["GET",
                  "HEAD"],
                  "quantity": 2
               },
               "forwardedValues": {
                  "cookies": {
                     "forward": "none"
                  },
                  "queryString": false
               },
               "minTTL": 300,
               "trustedSigners": {
                  "enabled": false,
                  "quantity": 0
               },
               "viewerProtocolPolicy": "redirect-to-https",
               "smoothStreaming": false
            },
            "origins": {
               "items": [{
                  "customOriginConfig": {
                     "hTTPSPort": 443,
                     "originProtocolPolicy": "http-only",
                     "hTTPPort": 80
                  },
                  "domainName": "myawsbucket.s3-website-us-east-2.amazonaws.com",
                  "id": "Web page origin"
               },
               {
                  "customOriginConfig": {
                     "hTTPSPort": 443,
                     "originProtocolPolicy": "http-only",
                     "hTTPPort": 80
                  },
                  "domainName": "myotherawsbucket.s3-website-us-west-2.amazonaws.com",
                  "id": "Images"
               }],
               "quantity": 2
            },
            "enabled": true,
            "cacheBehaviors": {
                  "allowedMethods": {
                     "items": ["GET",
                     "HEAD"],
                     "quantity": 2
                  },
                  "trustedSigners": {
                     "enabled": false,
                     "quantity": 0
                  },
                  "targetOriginId": "Web page origin",
                  "smoothStreaming": false,
                  "viewerProtocolPolicy": "redirect-to-https",
                  "minTTL": 300,
                  "forwardedValues": {
                     "cookies": {
                        "forward": "none"
```

```
                    },
                    "queryString": false
                },
                "pathPattern": "*.html"
            }],
            "quantity": 1
        },
        "priceClass": "PriceClass_All",
        "comment": "Added an origin and a cache behavior"
    }
},
"responseElements": {
    "eTag": "E2QWRUHEXAMPLE",
    "distribution": {
        "domainName": "d111111abcdef8.cloudfront.net",
        "status": "InProgress",
        "distributionConfig": {
        distributionConfig response omitted
        },
        "id": "EDFDVBD6EXAMPLE",
        "lastModifiedTime": "May 6, 2014 6:00:32 PM",
        "activeTrustedSigners": {
            "quantity": 0,
            "enabled": false
        },
        "inProgressInvalidationBatches": 0
    }
},
"requestID": "4e6b66f9-d548-11e3-a8a9-73e33example",
"eventID": "5ab02562-0fc5-43d0-b7b6-90293example"
},
{
    "eventVersion": "1.01",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "A1B2C3D4E5F6G7EXAMPLE",
        "arn": "arn:aws:iam::111122223333:user/smithj",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "smithj"
    },
    "eventTime": "2014-05-06T18:01:35Z",
    "eventName": "ListDistributions2014_01_31",
    "sourceIPAddress": "192.0.2.17",
    "userAgent": "aws-sdk-ruby/1.39.0 ruby/1.9.3 x86_64-linux",
    "requestParameters": null,
    "responseElements": null,
    "requestID": "52de9f97-d548-11e3-8fb9-4dad0example",
    "eventID": "eb91f423-6dd3-4bb0-a148-3cdfbexample"
},
{
    "eventVersion": "1.01",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "A1B2C3D4E5F6G7EXAMPLE",
        "arn": "arn:aws:iam::111122223333:user/smithj",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "smithj"
    },
    "eventTime": "2014-05-06T18:01:59Z",
    "eventName": "GetDistribution2014_01_31",
    "sourceIPAddress": "192.0.2.17",
    "userAgent": "aws-sdk-ruby/1.39.0 ruby/1.9.3 x86_64-linux",
    "requestParameters": {
        "id": "EDFDVBD6EXAMPLE"
```

```
      },
      "responseElements": null,
      "requestID": "497b3622-d548-11e3-8fb9-4dad0example",
      "eventID": "c32289c7-005a-46f7-9801-cba41example"
   },
   {
      "eventVersion": "1.01",
      "userIdentity": {
         "type": "IAMUser",
         "principalId": "A1B2C3D4E5F6G7EXAMPLE",
         "arn": "arn:aws:iam::111122223333:user/smithj",
         "accountId": "111122223333",
         "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
         "userName": "smithj"
      },
      "eventTime": "2014-05-06T18:02:27Z",
      "eventName": "CreateInvalidation2014_01_31",
      "sourceIPAddress": "192.0.2.17",
      "userAgent": "aws-sdk-ruby/1.39.0 ruby/1.9.3 x86_64-linux",
      "requestParameters": {
         "invalidationBatch": {
            "callerReference": "2014-05-06 64947",
            "paths": {
               "quantity": 3,
               "items": ["/images/new.jpg",
               "/images/logo.jpg",
               "/images/banner.jpg"]
            }
         },
         "distributionId": "EDFDVBD6EXAMPLE"
      },
      "responseElements": {
         "invalidation": {
            "createTime": "May 6, 2014 6:02:27 PM",
            "invalidationBatch": {
               "callerReference": "2014-05-06 64947",
               "paths": {
                  "quantity": 3,
                  "items": ["/images/banner.jpg",
                  "/images/logo.jpg",
                  "/images/new.jpg"]
               }
            },
            "status": "InProgress",
            "id": "ISRZ85EXAMPLE"
         },
         "location": "https://cloudfront.amazonaws.com/2014-01-31/distribution/
EDFDVBD6EXAMPLE/invalidation/ISRZ85EXAMPLE"
      },
      "requestID": "4e200613-d548-11e3-a8a9-73e33example",
      "eventID": "191ebb93-66b7-4517-a741-92b0eexample"
   },
   {
      "eventVersion": "1.01",
      "userIdentity": {
         "type": "IAMUser",
         "principalId": "A1B2C3D4E5F6G7EXAMPLE",
         "arn": "arn:aws:iam::111122223333:user/smithj",
         "accountId": "111122223333",
         "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
         "userName": "smithj"
      },
      "eventTime": "2014-05-06T18:03:08Z",
      "eventName": "ListCloudFrontOriginAccessIdentities2014_01_31",
      "sourceIPAddress": "192.0.2.17",
      "userAgent": "aws-sdk-ruby/1.39.0 ruby/1.9.3 x86_64-linux",
```

```
        "requestParameters": null,
        "responseElements": null,
        "requestID": "42ca4299-d548-11e3-8fb9-4dad0example",
        "eventID": "7aeb434f-eb55-4e2a-82d8-417d5example"
    }]
}
```

Amazon CloudWatch를 사용하여 CloudFront 모니터링

Amazon CloudFront는 Amazon CloudWatch와 통합되었으며, CloudFront 콘솔에서 그래프 세트로 표시되거나 CloudFront API 또는 CLI를 사용하여 액세스할 수 있는 배포당 10개의 운영 지표를 자동으로 게시합니다. 이러한 지표 중 일부는 CloudFront 배포에 적용되고 다른 지표는 Lambda @Edge 함수에 적용됩니다. 이러한 지표는 CloudWatch 할당량(이전에는 제한이라고 함)이 적용되지 않아 추가 비용이 발생하지 않습니다.

10개의 기본 지표 외에 추가 비용이 드는 지표를 추가로 활성화할 수 있습니다. 추가 지표는 CloudFront 배포에 적용되며, 각 배포에 대해 개별적으로 활성화해야 합니다. CloudWatch 요금에 관한 자세한 내용은 Amazon CloudWatch 요금을 참조하십시오.

이러한 지표를 보면 문제 해결, 추적 및 디버깅에 도움이 될 수 있습니다. 이러한 지표를 보는 방법은 CloudFront 콘솔의 모니터링 페이지를 참조하십시오. 특정 CloudFront 배포 또는 Lambda@Edge 함수의 활동에 대한 그래프를 보려면 그래프를 선택한 후 지표를 보도록 선택합니다.

CloudFront 콘솔이나 CloudWatch 콘솔, API 또는 CLI에서 이러한 지표에 다라 경보를 설정할 수도 있습니다(표준 CloudWatch 요금이 적용됩니다). 예를 들어, 응답의 HTTP 상태 코드가 `5xx` 범위에 있는 모든 최종 사용자 요청의 백분율을 나타내는 `5xxErrorRate` 지표에 따라 경보를 설정할 수 있습니다. 오류 발생률이 일정 시간 동안 일정 값에 도달하면(예: 연속 5분 동안 요청의 5%), 경보가 트리거됩니다. 경보를 만들 때 경보의 값과 시간 단위를 지정합니다. 자세한 내용은 경보 설정 (p. 418) 단원을 참조하십시오.

주제

- CloudFront 및 Lambda@Edge 지표 보기 (p. 416)
- 알림 수신을 위한 경보 설정 (p. 418)
- 데이터를 CSV 형식으로 다운로드 (p. 419)
- 프로그래밍 방식으로 CloudFront의 CloudWatch 지표 가져오기 (p. 421)

CloudFront 및 Lambda@Edge 지표 보기

CloudFront 콘솔에서 CloudFront 배포 및 Lambda @Edge 함수에 대한 운영 지표를 볼 수 있습니다. 이러한 지표를 보는 방법은 CloudFront 콘솔의 모니터링 페이지를 참조하십시오. 특정 CloudFront 배포 또는 Lambda@Edge 함수의 활동에 대한 그래프를 보려면 그래프를 선택한 후 지표를 보도록 선택합니다.

주제

- 기본 CloudFront 배포 지표 보기 (p. 416)
- 추가 CloudFront 배포 지표 보기 (p. 417)
- 기본 Lambda @Edge 함수 지표 보기 (p. 418)

기본 CloudFront 배포 지표 보기

다음 기본 지표는 모든 CloudFront 배포에 포함되며, 추가 비용은 들지 않습니다.

요청

모든 HTTP 메소드와 HTTP 및 HTTPS 둘 다에 대해 CloudFront에서 수신한 최종 사용자 요청의 총 수입니다.

다운로드된 바이트

`GET`, `HEAD`, `OPTIONS` 요청에 대해 최종 사용자가 다운로드한 총 바이트 수입니다.

업로드된 바이트

최종 사용자가 `POST` 및 `PUT` 요청을 사용하여 오리진에 CloudFront과 함께 업로드한 총 바이트 수입니다.

4xx 오류 발생률

응답의 HTTP 상태 코드가 `4xx`인 모든 최종 사용자 요청의 백분율입니다.

5xx 오류 발생률

응답의 HTTP 상태 코드가 `5xx`인 모든 최종 사용자 요청의 백분율입니다.

총 오류 발생률

응답의 HTTP 상태 코드가 `4xx` 또는 `5xx`인 모든 최종 사용자 요청의 백분율입니다.

이러한 지표는 CloudFront 콘솔의 모니터링 페이지에서 각 CloudFront 배포에 대해 그래프로 표시됩니다. 각 그래프에서 합계는 1분 간격으로 표시됩니다. 그래프를 보는 것 외에도 지표 보고서를 CSV 파일로 다운로드할 (p. 419) 수도 있습니다.

다음을 수행하여 그래프를 사용자 지정할 수 있습니다.

- 그래프에 표시된 정보의 시간 범위를 변경하려면 1h(1시간), 3h(3시간) 또는 다른 범위를 선택하거나 사용자 지정 범위를 지정합니다.
- CloudFront가 그래프의 정보를 업데이트하는 빈도를 변경하려면 새로 고침 아이콘 옆에 있는 아래쪽 화살표를 선택하고 새로 고침 간격을 선택합니다. 기본 새로 고침 빈도는 1분이지만 10초, 2분 또는 다른 옵션을 선택할 수 있습니다.

CloudWatch 콘솔에서 CloudFront 그래프를 보려면 대시보드에 추가를 선택합니다.

추가 CloudFront 배포 지표 보기

기본 지표 외에도 추가 비용이 드는 추가 지표를 활성화할 수 있습니다. 비용에 관한 자세한 내용은 Amazon CloudWatch 가격을 참조하십시오. 각 배포에 대해 다음 추가 지표를 개별적으로 활성화해야 합니다.

캐시 적중률

CloudFront이 캐시에서 콘텐츠를 제공한 캐시 가능한 모든 요청의 백분율입니다. HTTP `POST` 및 `PUT` 요청, 오류는 캐시 가능한 요청으로 간주되지 않습니다.

오리진 지연 시간

CloudFront에서 요청을 수신한 시점부터 CloudFront 캐시가 아닌 오리진에서 제공되는 요청에 대해 네트워크(최종 사용자 아님)에 대한 응답을 제공하기 시작한 시점까지 소요되는 총 시간입니다. 이것을 첫 번째 바이트 대기 시간 또는 time-to-first-byte라고도 합니다.

상태 코드별 오류 발생률

응답의 HTTP 상태 코드가 `4xx` 또는 `5xx` 범위의 특정 코드인 모든 최종 사용자 요청의 백분율입니다. 이 지표는 오류 코드 `401`, `403`, `404`, `502`, `503`, `504`에 모두 사용할 수 있습니다.

추가 지표를 활성화하려면

1. AWS Management 콘솔에 로그인하고 CloudFront 콘솔의 모니터링 페이지를 엽니다.

2. 추가 지표를 활성화할 배포를 선택한 다음 배포 지표 보기선택합니다.
3. 추가 지표 활성화를 선택합니다.
4. 추가 지표 활성화 창에서 활성화를 선택한 다음 저장을 선택합니다.

추가 지표를 활성화하면 그래프로 표시됩니다. 각 그래프에서 합계는 1분 간격으로 표시됩니다. 그래프를 보는 것 외에도 지표 보고서를 CSV 파일로 다운로드할 (p. 419) 수도 있습니다.

다음을 수행하여 그래프를 사용자 지정할 수 있습니다.

- 그래프에 표시된 정보의 시간 범위를 변경하려면 1h(1시간), 3h(3시간) 또는 다른 범위를 선택하거나 사용자 지정 범위를 지정합니다.
- CloudFront가 그래프의 정보를 업데이트하는 빈도를 변경하려면 새로 고침 아이콘 옆에 있는 아래쪽 화살표를 선택하고 새로 고침 간격을 선택합니다. 기본 새로 고침 빈도는 1분이지만 10초, 2분 또는 다른 옵션을 선택할 수 있습니다.

CloudWatch 콘솔에서 CloudFront 그래프를 보려면 대시보드에 추가를 선택합니다.

기본 Lambda @Edge 함수 지표 보기

다음 기본 지표는 모든 Lambda @Edge 함수에 포함되며, 추가 비용은 들지 않습니다.

- Lambda@Edge의 5xx 오류 발생률
- Lambda 실행 오류
- 잘못된 Lambda 응답
- Lambda 제한

이러한 지표는 CloudFront콘솔의 모니터링 페이지에서 각 Lambda@Edge 함수에 대해 그래프로 표시됩니다. 그래프에는 호출 수, 오류 수, 제한 수 등이 포함되어 있습니다. 각 그래프에서 합계는 1분 간격으로 표시되며 AWS 리전별로 그룹화됩니다.

조사할 오류가 급증할 경우 함수를 선택한 다음 AWS 리전별로 로그 파일을 확인하여, 어떤 함수가 문제를 야기했는지 및 어느 AWS 리전에서 발생했는지 파악할 수 있습니다. Lambda@Edge 오류 문제 해결에 대한 자세한 내용은 다음 단원을 참조하십시오.

- 실패 유형을 확인하는 방법 (p. 308)
- AWS에서 콘텐츠 전송 디버깅을 위한 4단계

다음을 수행하여 그래프를 사용자 지정할 수 있습니다.

- 그래프에 표시된 정보의 시간 범위를 변경하려면 1h(1시간), 3h(3시간) 또는 다른 범위를 선택하거나 사용자 지정 범위를 지정합니다.
- CloudFront가 그래프의 정보를 업데이트하는 빈도를 변경하려면 새로 고침 아이콘 옆에 있는 아래쪽 화살표를 선택하고 새로 고침 간격을 선택합니다. 기본 새로 고침 빈도는 1분이지만 10초, 2분 또는 다른 옵션을 선택할 수 있습니다.

CloudWatch 콘솔에서 그래프를 보려면 대시보드에 추가를 선택합니다.

알림 수신을 위한 경보 설정

CloudFront 콘솔에서 특정 CloudFront 지표에 따라 Amazon Simple Notification Service(Amazon SNS)를 통해 알리도록 경보를 설정할 수 있습니다. CloudFront콘솔의 경보 페이지에서 경보를 설정할 수 있습니다.

콘솔에서 경보를 만들 때 다음 값을 지정합니다.

지표

경보를 생성 중인 지표입니다.

배포

경보를 생성 중인 CloudFront 배포입니다.

경보 이름

경보의 이름입니다.

알림 전송 대상

이 지표가 경보를 트리거할 경우 알림을 보낼 Amazon SNS 주제입니다.

`<metric> <operator> <value>`일 때 항상

CloudWatch에서 경보를 트리거하고 Amazon SNS topic으로 알림을 전송하는 시점을 지정합니다. 예를 들어, `5xx` 오류 발생률이 1퍼센트를 넘는 경우 알림을 받으려면 다음과 같이 지정할 수 있습니다.

평균이 5xxErrorRate일 때 항상 `> 1`

값 지정 시 다음에 유의하십시오.

- 문장 부호 없이 숫자만 입력합니다. 예를 들어, 천을 지정하려면 `1000`을 입력합니다.
- `4xx`, `5xx` 및 총 오류 발생률은 지정하는 값이 백분율입니다.
- 요청, 다운로드한 바이트 및 업로드한 바이트는 지정하는 값이 단위입니다. 예를 들면 1073742000 바이트입니다.

`<time period>`의 최소 `<number>` 연속 기간

CloudWatch에서 경보를 트리거하기 전에 지표가 충족해야 할 조건을 지정한 기간 동안의 연속된 기간 수로 지정합니다. 값을 선택할 때 일시적 문제는 경보가 발생하지 않지만 지속되거나 실제 문제는 경보가 발생하는 값을 적절하게 조정해야 합니다.

데이터를 CSV 형식으로 다운로드

CloudFront 배포에 대한 CloudWatch 지표 데이터를 CSV 형식으로 다운로드할 수 있습니다. CloudFront콘솔에서 특정 배포에 대한 배포 지표를 볼 때 데이터를 다운로드할 수 있습니다.

보고서에 대한 정보

보고서의 처음 몇 줄에는 다음 정보가 포함됩니다.

Version

CloudFront 보고 버전입니다.

Report

보고서의 이름입니다.

DistributionID

보고서를 실행한 배포의 ID입니다.

StartDateUTC

보고서 실행 기간에 해당하는 날짜 범위의 시작 일시를 협정 세계시(UTC)로 표시합니다.

EndDateUTC

보고서 실행 기간에 해당하는 날짜 범위의 종료 일시를 협정 세계시(UTC)로 표시합니다.

GeneratedTimeUTC

보고서를 실행 날짜 및 시간을 협정 세계시(UTC)로 표시합니다.

Granularity

보고서 각 행의 기간입니다(예: `ONE_MINUTE`).

CloudWatch Metrics 보고서의 데이터

이 보고서에는 다음 값이 포함됩니다.

DistributionID

보고서를 실행한 배포의 ID입니다.

FriendlyName

배포의 대체 도메인 이름(CNAME)입니다(있는 경우). 배포에 대체 도메인 이름이 없으면 목록에 배포에 대한 원래 도메인 이름이 포함됩니다.

TimeBucket

데이터가 적용되는 시간 또는 날짜를 협정 세계시(UTC)로 표시합니다.

Requests

해당 기간 동안의 전체 HTTP 상태 코드(예: `200`, `404` 등)와 전체 메소드(예: `GET`, `HEAD`, `POST` 등)의 총 요청 수입니다.

BytesDownloaded

해당 기간 동안 지정한 배포에 대해 최종 사용자가 다운로드한 바이트 수입니다.

BytesUploaded

해당 기간 동안 지정한 배포에 대해 최종 사용자가 업로드한 바이트 수입니다.

TotalErrorRatePct

해당 기간 동안 지정한 배포에 대해 HTTP 상태 코드가 `4xx` 또는 `5xx` 오류였던 요청의 백분율입니다.

4xxErrorRatePct

해당 기간 동안 지정한 배포에 대해 HTTP 상태 코드가 `4xx` 오류였던 요청의 백분율입니다.

5xxErrorRatePct

해당 기간 동안 지정한 배포에 대해 HTTP 상태 코드가 `5xx` 오류였던 요청의 백분율입니다.

배포에 대해 추가 지표를 활성화 (p. 417)한 경우에는 보고서에 다음과 같은 추가 값도 포함됩니다.

401ErrorRatePct

해당 기간 동안 지정한 배포에 대해 HTTP 상태 코드가 `401` 오류였던 요청의 백분율입니다.

403ErrorRatePct

해당 기간 동안 지정한 배포에 대해 HTTP 상태 코드가 `403` 오류였던 요청의 백분율입니다.

404ErrorRatePct

해당 기간 동안 지정한 배포에 대해 HTTP 상태 코드가 `404` 오류였던 요청의 백분율입니다.

502ErrorRatePct

해당 기간 동안 지정한 배포에 대해 HTTP 상태 코드가 `502` 오류였던 요청의 백분율입니다.

503ErrorRatePct

해당 기간 동안 지정한 배포에 대해 HTTP 상태 코드가 `503` 오류였던 요청의 백분율입니다.

504ErrorRatePct

해당 기간 동안 지정한 배포에 대해 HTTP 상태 코드가 `504` 오류였던 요청의 백분율입니다.

OriginLatency

CloudFront에서 요청을 수신한 시점부터 CloudFront 캐시가 아닌 오리진에서 제공되는 요청에 대해 네트워크(최종 사용자 아님)에 대한 응답을 제공하기 시작한 시점까지 소요되는 총 시간(밀리초)입니다. 이것을 첫 번째 바이트 대기 시간 또는 time-to-first-byte라고도 합니다.

CacheHitRate

CloudFront이 캐시에서 콘텐츠를 제공한 캐시 가능한 모든 요청의 백분율입니다. HTTP `POST` 및 `PUT` 요청, 오류는 캐시 가능한 요청으로 간주되지 않습니다.

프로그래밍 방식으로 CloudFront의 CloudWatch 지표 가져오기

Amazon CloudWatch API 또는 CLI를 사용하여, 빌드한 프로그램이나 애플리케이션의 CloudFront 지표를 가져올 수 있습니다. 원시 데이터를 사용하여 사용자 지정 대시보드, 사용자 고유의 경고 도구 등을 만들 수 있습니다. CloudWatch API에서 CloudFront 지표를 가져오려면 각 지표의 특정 값과 유형을 알아야 합니다.

주제

- 모든 CloudFront 지표에 대한 값 (p. 421)
- 특정 CloudFront 지표에 대한 값 (p. 421)

모든 CloudFront 지표에 대한 값

다음 값은 모든 CloudFront 지표에 적용됩니다.

`Namespace`

`Namespace`의 값은 항상 `AWS/CloudFront`입니다.

차원

각 CloudFront 지표에는 다음과 같은 두 가지 차원이 있습니다.

`DistributionId`

`DistributionId`의 값은 지표를 가져오려는 CloudFront 배포의 ID입니다.

`Region`

`Region`의 값은 항상 `Global`입니다. CloudFront은 글로벌 서비스이기 때문입니다.

특정 CloudFront 지표에 대한 값

다음 목록의 정보를 사용하여 CloudWatch API에서 특정 CloudFront 지표에 대한 세부 정보를 얻을 수 있습니다. 이러한 지표 중 일부는 배포에 대한 추가 지표를 활성화한 경우에만 사용할 수 있습니다.

Note

각 지표에 대해 하나의 통계, `Average` 또는 `Sum`만 적용할 수 있습니다. 다음 목록은 해당 지표에 적용할 수 있는 통계를 지정합니다.

4xx 오류 발생률

응답의 HTTP 상태 코드가 `4xx`인 모든 최종 사용자 요청의 백분율입니다.

- 지표 이름: `4xxErrorRate`
- 유효한 통계: `Average`
- 단위: `Percent`

401 오류 발생률

응답의 HTTP 상태 코드가 `401`인 모든 최종 사용자 요청의 백분율입니다. 이 지표를 가져오려면 먼저 추가 지표를 활성화 (p. 417)해야 합니다.

- 지표 이름: `401ErrorRate`
- 유효한 통계: `Average`
- 단위: `Percent`

403 오류 발생률

응답의 HTTP 상태 코드가 `403`인 모든 최종 사용자 요청의 백분율입니다. 이 지표를 가져오려면 먼저 추가 지표를 활성화 (p. 417)해야 합니다.

- 지표 이름: `403ErrorRate`
- 유효한 통계: `Average`
- 단위: `Percent`

404 오류 발생률

응답의 HTTP 상태 코드가 `404`인 모든 최종 사용자 요청의 백분율입니다. 이 지표를 가져오려면 먼저 추가 지표를 활성화 (p. 417)해야 합니다.

- 지표 이름: `404ErrorRate`
- 유효한 통계: `Average`
- 단위: `Percent`

5xx 오류 발생률

응답의 HTTP 상태 코드가 `5xx`인 모든 최종 사용자 요청의 백분율입니다.

- 지표 이름: `5xxErrorRate`
- 유효한 통계: `Average`
- 단위: `Percent`

502 오류 발생률

응답의 HTTP 상태 코드가 `502`인 모든 최종 사용자 요청의 백분율입니다. 이 지표를 가져오려면 먼저 추가 지표를 활성화 (p. 417)해야 합니다.

- 지표 이름: `502ErrorRate`
- 유효한 통계: `Average`
- 단위: `Percent`

503 오류 발생률

응답의 HTTP 상태 코드가 `503`인 모든 최종 사용자 요청의 백분율입니다. 이 지표를 가져오려면 먼저 추가 지표를 활성화 (p. 417)해야 합니다.

- 지표 이름: `503ErrorRate`
- 유효한 통계: `Average`
- 단위: `Percent`

504 오류 발생률

응답의 HTTP 상태 코드가 `504`인 모든 최종 사용자 요청의 백분율입니다. 이 지표를 가져오려면 먼저 추가 지표를 활성화 (p. 417)해야 합니다.

- 지표 이름: `504ErrorRate`
- 유효한 통계: `Average`
- 단위: `Percent`

다운로드된 바이트

`GET`, `HEAD`, `OPTIONS` 요청에 대해 최종 사용자가 다운로드한 총 바이트 수입니다.

- 지표 이름: `BytesDownloaded`
- 유효한 통계: `Sum`
- 단위: `None`

업로드된 바이트

최종 사용자가 `POST` 및 `PUT` 요청을 사용하여 오리진에 CloudFront과 함께 업로드한 총 바이트 수입니다.

- 지표 이름: `BytesUploaded`
- 유효한 통계: `Sum`
- 단위: `None`

캐시 적중률

CloudFront이 캐시에서 콘텐츠를 제공한 캐시 가능한 모든 요청의 백분율입니다. HTTP `POST` 및 `PUT` 요청, 오류는 캐시 가능한 요청으로 간주되지 않습니다. 이 지표를 가져오려면 먼저 추가 지표를 활성화 (p. 417)해야 합니다.

- 지표 이름: `CacheHitRate`
- 유효한 통계: `Average`
- 단위: `Percent`

오리진 지연 시간

CloudFront에서 요청을 수신한 시점부터 CloudFront 캐시가 아닌 오리진에서 제공되는 요청에 대해 네트워크(최종 사용자 아님)에 대한 응답을 제공하기 시작하는 시점까지 소요되는 총 시간(밀리초)입니다. 이것을 첫 번째 바이트 대기 시간 또는 time-to-first-byte라고도 합니다. 이 지표를 가져오려면 먼저 추가 지표를 활성화 (p. 417)해야 합니다.

- 지표 이름: `OriginLatency`
- 유효한 통계: `Percentile`
- 단위: `Milliseconds`

> **Note**
>
> CloudWatch API에서 `Percentile` 통계를 확인하려면 `Statistics`가 아닌 `ExtendedStatistics` 파라미터를 사용하십시오. 자세한 내용은 Amazon CloudWatch API Reference의 GetMetricStatistics 또는 AWS SDK 관련 참조 설명서를 참조하십시오.

요청

모든 HTTP 메소드와 HTTP 및 HTTPS 둘 다에 대해 CloudFront에서 수신한 최종 사용자 요청의 총 수입니다.

- 지표 이름: `Requests`
- 유효한 통계: `Sum`
- 단위: `None`

총 오류 발생률

응답의 HTTP 상태 코드가 `4xx` 또는 `5xx`인 모든 최종 사용자 요청의 백분율입니다.

- 지표 이름: `TotalErrorRate`
- 유효한 통계: `Average`
- 단위: `Percent`

Amazon CloudFront 배포에 태그 지정

태그란 AWS 리소스를 식별하고 정리할 때 사용할 수 있는 단어 또는 문구입니다. 각 리소스에 태그를 여러 개 추가할 수 있고, 각 태그는 사용자가 정의한 키와 값을 포함할 수 있습니다. 예를 들어, 키는 "도메인"이고 값은 "example.com"일 수 있습니다. 추가하는 태그에 따라 리소스를 검색하고 필터링할 수 있습니다.

다음은 이 기능이 CloudFront에서 태그로 작업하는 데 유용하게 사용될 수 있는 방법을 보여 주는 두 가지 예제입니다.

- 태그를 사용하여 다양한 범주에서 청구 정보를 추적합니다. CloudFront 배포 또는 기타 AWS 리소스(Amazon EC2 인스턴스 또는 Amazon S3 버킷 등)에 태그를 적용하고 태그를 활성화하면 AWS는 사용 내역 및 비용을 활성 태그 기준으로 집계한 비용 할당 보고서를 CSV 파일 형식으로 생성합니다. 비즈니스 범주를 나타내는 태그(예: 비용 센터, 애플리케이션 이름 또는 소유자)를 적용하여 여러 서비스에 대한 비용을 정리할 수 있습니다. 비용 할당 태그 사용에 대한 자세한 내용은 AWS Billing and Cost Management 사용 설명서의 비용 할당 태그 사용을 참조하십시오.
- 태그를 사용하여 CloudFront 배포에 대한 태그 기반 권한을 적용합니다. 자세한 내용은 태그 기반 정책 (p. 432)단원을 참조하십시오.

다음을 참조하십시오.

- 웹 및 RTMP 배포에는 태그를 지정할 수 있지만 원본 액세스 ID 또는 배포에는 태그 지정이 불가능합니다.
- 현재 CloudFront에 대해서는 Tag Editor 또는 리소스 그룹이 지원되지 않습니다.

현재 배포에 추가할 수 있는 태그의 최대 수는 할당량 (p. 447) 단원을 참조하십시오. 할당량(이전에는 제한이라고 함)을 더 높여 달라고 요청하려면 AWS 지원 센터에서 사례를 생성합니다.

CloudFront API, AWS CLI, SDK 및 Windows PowerShell용 AWS 도구를 사용하여 리소스에 태그를 적용할 수도 있습니다. 자세한 내용은 다음 설명서를 참조하십시오.

- CloudFront API – Amazon CloudFront API Reference에서 다음 작업을 참조하십시오.
  - ListTagsForResource
  - TagResource
  - UntagResource
- AWS CLI – AWS CLI Command Reference의 cloudfront 참조
- SDK – AWS 설명서 페이지에서 해당 SDK 설명서 참조
- Windows PowerShell용 도구 – PowerShell용 AWS 도구 Cmdlet Reference의 Amazon CloudFront 참조

주제

- 태그 제한 (p. 424)
- 배포에 태그 추가, 편집, 삭제 (p. 425)

태그 제한

태그에 적용되는 기본 제한은 다음과 같습니다.

- 리소스당 최대 태그 수 – 50개
- 최대 키 길이 – 유니코드 128자
- 최대 값 길이 –- 유니코드 256자
- 키 및 값의 유효값 – a-z, A-Z, 0-9, 공백 및 특수 문자 _ . : / = + - 및 @
- 태그 키와 값은 대/소문자를 구분합니다

- 키 접두사로 `aws:`를 사용하지 마세요. AWS 전용입니다.

배포에 태그 추가, 편집, 삭제

다음 절차는 CloudFront 콘솔에서 배포에 태그를 추가, 편집, 삭제하는 방법을 설명합니다.

배포에 태그를 추가, 편집 또는 삭제하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/cloudfront/에서 CloudFront 콘솔을 엽니다.
2. 업데이트하려는 배포의 ID를 선택합니다.
3. [Tags] 탭을 선택합니다.
4. 태그 추가 또는 편집를 선택합니다.
5. 태그 추가 또는 편집 페이지에서 다음 작업을 수행할 수 있습니다.

   태그 추가

   키를 입력하고, 원한다면 태그 값도 입력합니다.

   태그 편집

   키, 값 또는 둘 다 변경합니다. 태그의 값은 삭제할 수 있지만 키는 필요합니다.

   태그 삭제

   값 필드 오른쪽에 있는 X를 선택합니다.
6. [Save]를 선택합니다.

Amazon CloudFront의 보안

AWS에서는 클라우드 보안을 가장 중요하게 생각합니다. AWS 고객은 보안에 매우 보안에 민감한 조직의 요구 사항에 부합하도록 구축된 데이터 센터 및 네트워크 아키텍처의 혜택을 누릴 수 있습니다.

보안은 AWS와 귀하의 공동 책임입니다. 책임 분담 모델은 이 사항을 클라우드 내 보안 및 클라우드의 보안으로 설명합니다.

- 클라우드의 보안 – AWS는 AWS 클라우드에서 AWS 서비스를 실행하는 인프라를 보호합니다. AWS는 또한 안전하게 사용할 수 있는 서비스를 제공합니다. 타사 감사원은 정기적으로 AWS 규제 준수 프로그램의 일환으로 보안 효과를 테스트하고 검증합니다. Amazon CloudFront에 적용되는 규정 준수 프로그램에 대한 자세한 내용은 규정 준수 프로그램 제공 AWS 범위 내 서비스를 참조하십시오.
- 클라우드 내 보안 – 귀하의 책임은 귀하가 사용하는 AWS 서비스에 의해 결정됩니다. 또한 데이터의 민감도, 조직의 요구 사항 및 관련 법률 및 규정을 비롯한 기타 요소에 대해서도 책임이 있습니다.

이 설명서는 CloudFront 사용 시 책임 분담 모델을 적용하는 방법을 이해하는 데 도움이 됩니다. 다음 항목에서는 보안 및 규정 준수 목표를 충족하도록 CloudFront를 구성하는 방법을 보여줍니다. 또한 CloudFront 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 AWS 제품을 사용하는 방법을 배웁니다.

주제

- Amazon CloudFront의 데이터 보호 (p. 426)
- CloudFront의 Identity and Access Management (p. 428)
- Amazon CloudFront의 로깅 및 모니터링 (p. 444)
- Amazon CloudFront 규정 준수 확인 (p. 444)
- Amazon CloudFront의 복원성 (p. 446)
- Amazon CloudFront의 인프라 보안 (p. 446)

Amazon CloudFront의 데이터 보호

Amazon CloudFront는 AWS 공동 책임 모델을 준수하며, 여기에는 데이터 보호 관련 규정 및 지침이 포함됩니다. AWS는 모든 AWS 서비스를 실행하는 글로벌 인프라를 보호할 책임을 갖습니다. AWS는 고객 콘텐츠 및 개인 데이터의 처리를 위한 보안 구성 제어 등 이 인프라에서 호스팅되는 데이터에 대한 제어권을 유지합니다. 데이터 컨트롤러 또는 데이터 처리자의 역할을 담당하는 AWS 고객과 APN 파트너는 AWS Cloud에 올린 모든 개인 데이터에 대한 책임을 갖습니다.

데이터 보호를 위해 AWS 계정 자격 증명을 보호하고 AWS Identity and Access Management (IAM)을 사용하여 개별 사용자 계정을 설정하여 각 사용자에게 직무를 수행하는 데 필요한 권한만 부여하는 것이 좋습니다. 또한 다음과 같은 방법으로 데이터를 보호하는 것이 좋습니다.

- 각 계정마다 멀티 팩터 인증(MFA)을 사용합니다.
- SSL/TLS를 사용하여 AWS 리소스와 통신합니다.
- AWS CloudTrail로 API 및 사용자 활동 로깅을 설정합니다.
- AWS 암호화 솔루션을 AWS 서비스 내의 모든 기본 보안 컨트롤과 함께 사용합니다.
- Amazon S3에 저장된 개인 데이터를 검색하고 보호하는 데 도움이 되는 Amazon Macie와 같은 고급 관리형 보안 서비스를 사용합니다.

이름 필드와 같은 자유 형식 필드에 고객 계정 번호와 같은 중요 식별 정보를 절대 입력하지 마십시오. 여기에는 CloudFront 또는 기타 AWS 제품에서 콘솔, API, AWS CLI 또는 AWS SDK를 사용하여 작업하는 경우

가 포함됩니다. CloudFront 또는 기타 서비스에 입력하는 모든 데이터는 진단 로그에 포함하기 위해 선택될 수 있습니다. 외부 서버에 URL을 제공할 때 해당 서버에 대한 요청을 검증하기 위해 자격 증명 정보를 URL에 포함시키지 마십시오.

데이터 보호에 관한 자세한 내용은 AWS 보안 블로그에서 AWS 공동 책임 모델 및 GDPR 블로그 게시물을 참조하십시오.

Amazon CloudFront는 제공하는 콘텐츠의 보안을 유지하는 데 사용할 수 있는 몇 가지 옵션을 제공합니다.

- HTTPS 연결 구성
- 전송 중 데이터를 암호화하도록 필드 수준 암호화 구성
- 특정 사람이나 특정 영역의 사람만 볼 수 있도록 콘텐츠에 대한 액세스 제한

다음 주제에서는 이러한 옵션에 대해 자세히 설명합니다.

주제

- 전송 중 데이터 암호화 (p. 427)
- 저장 데이터 암호화 (p. 427)
- 콘텐츠에 대한 액세스 제한 (p. 427)

전송 중 데이터 암호화

전송 중 데이터를 암호화하려면 최종 사용자가 HTTPS를 사용하여 파일을 요청하도록 Amazon CloudFront를 구성합니다. 이렇게 하면 CloudFront가 최종 사용자와 통신할 때 연결이 암호화됩니다. 또한 CloudFront가 오리진의 객체를 받을 때 HTTPS를 사용하도록 구성할 수 있습니다. 이렇게 하면 CloudFront가 오리진과 통신할 때 연결이 암호화됩니다.

자세한 내용은 CloudFront에서 HTTPS 사용 (p. 92) 단원을 참조하십시오.

필드 레벨 암호화는 HTTPS와 함께 추가 보안 레이어를 추가하여 시스템 처리 전체에서 특정 데이터를 보호하고 특정 애플리케이션만 이를 볼 수 있도록 합니다. CloudFront에서 필드 수준 암호화를 구성하여 사용자가 제출한 중요한 정보를 웹 서버에 안전하게 업로드할 수 있습니다. 클라이언트가 제공하는 중요한 정보는 사용자에게 더 가까운 엣지에서 암호화됩니다. 전체 애플리케이션 스택에서 암호화된 상태로 유지되므로 데이터가 필요하고 이를 해독할 자격 증명이 있는 애플리케이션만 이 작업을 수행할 수 있습니다.

자세한 내용은 필드 레벨 암호화 사용으로 민감한 데이터 보호 (p. 185) 단원을 참조하십시오.

저장 데이터 암호화

CloudFront는 엣지 위치 POP(상호 접속 위치)용으로 암호화된 SSD 및 Regional Edge Caches(REC)용으로 암호화된 EBS 볼륨을 사용합니다.

콘텐츠에 대한 액세스 제한

인터넷을 통해 콘텐츠를 배포하는 많은 회사에서는 일부 사용자를 대상으로 하는 콘텐츠, 문서, 비즈니스 데이터 또는 미디어 스트림에 대한 액세스를 제한하고자 합니다. Amazon CloudFront를 사용하여 이 콘텐츠를 안전하게 제공하려면 다음 중 하나 이상을 수행하면 됩니다.

서명된 URL 또는 쿠키 사용

서명된 URL 또는 서명된 쿠키를 사용하여 CloudFront를 통해 이 프라이빗 콘텐츠를 제공함으로써 선택된 사용자(예: 유료 사용자)를 대상으로 하는 콘텐츠에 대한 액세스를 제한할 수 있습니다. 자세한 내용은 서명된 URL과 서명된 쿠키를 사용하여 프라이빗 콘텐츠 제공 (p. 115) 단원을 참조하십시오.

Amazon S3 버킷의 콘텐츠에 대한 액세스 제한

예를 들어 CloudFront 서명된 URL 또는 서명된 쿠키를 사용해 콘텐츠에 대한 액세스를 제한하는 경우 사용자가 파일에 대한 직접 URL을 사용하여 파일을 보는 것도 원치 않을 것입니다. 대신, 보호가 작동하도록 CloudFront URL을 사용해야만 파일에 액세스하도록 허용할 수 있습니다.

Amazon S3 버킷을 CloudFront 배포의 오리진으로 사용하는 경우 OAI(원본 액세스 ID)를 설정하여 콘텐츠에 대한 직접 액세스를 관리할 수 있습니다. 원본 액세스 ID는 배포와 연결할 수 있는 특별한 CloudFront 사용자 자격 증명이므로 전체 또는 일부 Amazon S3 콘텐츠를 보호할 수 있습니다. 이 작업을 구성하는 방법에 대한 자세한 내용은 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오.

AWS WAF 웹 ACL 사용

웹 애플리케이션 방화벽 서비스인 AWS WAF를 사용하여 웹 액세스 제어 목록(웹 ACL)을 만들어 콘텐츠에 대한 액세스를 제한할 수 있습니다. 요청이 허용되는 IP 주소나 쿼리 문자열의 값으로부터 지정하는 조건 등 지정하는 조건에 따라, CloudFront는 요청된 콘텐츠나 HTTP 403 상태 코드(금지됨)로 요청에 응답합니다. 자세한 내용은 AWS AWS WAF를 사용하여 콘텐츠에 대한 액세스 제어 (p. 182) 단원을 참조하십시오.

지리적 제한 사용

지리적 차단이라고도 하는 지리적 제한을 사용하여 특정 지리적 위치에 있는 사용자가 CloudFront 배포를 통해 제공한 콘텐츠에 액세스하는 것을 차단할 수 있습니다. 지리적 제한을 구성할 때 몇 가지 옵션을 선택할 수 있습니다. 자세한 내용은 콘텐츠의 지리적 배포 제한 (p. 183) 단원을 참조하십시오.

CloudFront의 Identity and Access Management

CloudFront 리소스에서 배포 만들기 또는 객체 무효화와 같은 작업을 수행하려면 AWS Identity and Access Management(IAM)에서 사용자가 승인된 AWS 사용자임을 인증해야 합니다. CloudFront 콘솔을 사용하는 경우 AWS 사용자 이름과 암호를 제공하여 자격 증명을 인증합니다. 프로그래밍 방식으로 CloudFront에 액세스하는 경우 애플리케이션에서 액세스 키를 사용하거나 요청에 서명하여 자격 증명을 인증합니다.

자격 증명을 인증한 후 IAM는 사용자가 작업을 수행하고 리소스에 액세스할 권한이 있는지 확인하여 AWS에 대한 액세스를 제어합니다. 계정 관리자인 경우 IAM을 사용하여 계정과 관련된 리소스에 대한 다른 사용자의 액세스를 제어할 수 있습니다.

이 장에서는 AWS Identity and Access Management(IAM) 및 CloudFront를 사용하여 리소스를 보호하는 방법을 설명합니다.

주제

- 인증 (p. 428)
- 액세스 제어 (p. 429)

인증

다음과 같은 ID 유형으로 AWS에 액세스할 수 있습니다.

- AWS 계정 루트 사용자 – AWS 계정을 처음 생성할 때는 해당 계정의 모든 AWS 서비스 및 리소스에 대한 완전한 액세스 권한이 있는 SSO(Single Sign-In) ID로 시작합니다. 이 자격 증명은 AWS 계정 루트 사용자라고 하며, 계정을 생성할 때 사용한 이메일 주소와 암호로 로그인하여 액세스합니다. 일상적인 작업은 물론 관리 작업에도 루트 사용자를 사용하지 않는 것이 좋습니다. 대신 IAM 사용자를 처음 생성할 때만 루트 사용자를 사용하는 모범 사례를 준수하십시오. 그런 다음 루트 사용자 자격 증명을 안전하게 보관해 두고 몇 가지 계정 및 서비스 관리 작업을 수행할 때만 해당 자격 증명을 사용합니다.
- IAM 사용자 – IAM 사용자는 특정 사용자 지정 권한(예: CloudFront에서 웹 배포을 생성할 권한)이 있는 AWS 계정 내 자격 증명입니다. IAM 사용자 이름과 암호를 사용하여 AWS Management 콘솔, AWS 토론 포럼 또는 AWS Support Center 같은 보안 AWS 웹 페이지에 로그인할 수 있습니다.

사용자 이름과 암호 외에도 각 사용자에 대해 액세스 키를 생성할 수 있습니다. 여러 SDK 중 하나를 통해 또는 AWS Command Line Interface(CLI)를 사용하여 AWS 제품에 프로그래밍 방식으로 액세스할 때 이러한 키를 사용할 수 있습니다. SDK 및 CLI 도구는 액세스 키를 사용하여 암호화 방식으로 요청에 서명합니다. AWS 도구를 사용하지 않는 경우 요청에 직접 서명해야 합니다. CloudFront 지원는 인바운드 API 요청을 인증하기 위한 프로토콜인 서명 버전 4를 지원합니다. 요청 인증에 대한 자세한 내용은 AWS General Reference의 서명 버전 4 서명 프로세스 단원을 참조하십시오.

- IAM 역할 – IAM 역할은 특정 권한을 가진 계정에서 생성할 수 있는 IAM 자격 증명입니다. IAM 역할은 AWS에서 자격 증명으로 할 수 있는 것과 할 수 없는 것을 결정하는 권한 정책을 포함하는 AWS 자격 증명이라는 점에서 IAM 사용자와 유사합니다. 그러나 역할은 한 사람과만 연관되지 않고 해당 역할이 필요한 사람이라면 누구든지 맡을 수 있어야 합니다. 또한 역할에는 그와 연관된 암호 또는 액세스 키와 같은 표준 장기 자격 증명이 없습니다. 대신에 역할을 수임한 사람에게는 해당 역할 세션을 위한 임시 보안 자격 증명이 제공됩니다. 임시 자격 증명이 있는 IAM 역할은 다음과 같은 상황에서 유용합니다.

  - 연합된 사용자 액세스 – IAM 사용자를 생성하는 대신 AWS Directory Service의 기존 ID, 엔터프라이즈 사용자 디렉터리 또는 웹 ID 공급자를 사용할 수 있습니다. 이 사용자를 연합된 사용자라고 합니다. AWS에서는 ID 공급자를 통해 액세스가 요청되면 연합된 사용자에게 역할을 할당합니다. 연합된 사용자에 대한 자세한 내용은 IAM 사용 설명서의 연합된 사용자 및 역할을 참조하십시오.

  - AWS 서비스 액세스 – 서비스 역할은 서비스가 사용자를 대신하여 사용자 계정에서 작업을 수행하기 위해 수임하는 IAM 역할입니다. 일부 AWS 서비스 환경을 설정할 때 서비스에서 맡을 역할을 정의해야 합니다. 이 서비스 역할에는 서비스가 AWS 리소스에 액세스하는 데 필요한 모든 권한이 포함되어야 합니다. 서비스 역할은 서비스마다 다르지만 해당 서비스에 대한 문서화된 요구 사항을 충족하는 한 대부분의 경우 권한을 선택할 수 있습니다. 서비스 역할은 해당 계정 내 액세스 권한만 제공하며 다른 계정의 서비스에 대한 액세스 권한을 부여하는 데 사용될 수 없습니다. IAM 내에서 서비스 역할을 생성, 수정 및 삭제할 수 있습니다. 예를 들어 Amazon Redshift에서 사용자 대신 Amazon S3 버킷에 액세스하도록 허용하는 역할을 생성한 후 해당 버킷에 있는 데이터를 Amazon Redshift 클러스터로 로드할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 AWS 서비스에 대한 권한을 위임할 역할 생성을 참조하십시오.

  - Amazon EC2에서 실행 중인 애플리케이션 – IAM 역할을 사용하여 EC2 인스턴스에서 실행되고 AWS CLI 또는 AWS API 요청을 수행하는 애플리케이션의 임시 자격 증명을 관리할 수 있습니다. 이는 EC2 인스턴스 내에 액세스 키를 저장할 때 권장되는 방법입니다. EC2 인스턴스에 AWS 역할을 할당하고 해당 역할을 모든 애플리케이션에서 사용할 수 있도록 하려면 인스턴스에 연결된 인스턴스 프로파일을 생성합니다. 인스턴스 프로파일에는 역할이 포함되어 있으며 EC2 인스턴스에서 실행되는 프로그램이 임시 자격 증명을 얻을 수 있습니다. 자세한 내용은 IAM 사용 설명서의 IAM 역할을 사용하여 Amazon EC2 인스턴스에서 실행되는 애플리케이션에 권한 부여를 참조하십시오.

액세스 제어

CloudFront 리소스를 생성, 업데이트, 삭제 또는 나열하려면 작업을 수행할 수 있는 권한이 필요하며 해당 리소스에 액세스할 수 있는 권한이 필요합니다. 또한 프로그래밍 방식으로 작업을 수행하려면 유효한 액세스 키가 필요합니다.

다음 단원에서는 CloudFront에 대한 권한을 관리하는 방법에 대해 설명합니다.

- CloudFront 리소스에 대한 액세스 권한 관리 개요 (p. 430)
- CloudFront에 대한 자격 증명 기반 정책(IAM 정책) 사용 (p. 434)
- CloudFront API 권한: 작업, 리소스 및 조건 참조 (p. 439)

CloudFront 리소스에 대한 액세스 권한 관리 개요

모든 AWS 리소스는 AWS 계정의 소유이고, 리소스 생성 또는 리소스 액세스 권한은 권한 정책에 따라 결정됩니다.

> **Note**
>
> 계정 관리자 또는 관리자 사용자는 관리자 권한이 있는 사용자입니다. 관리자에 대한 자세한 내용은 IAM 사용 설명서의 IAM 모범 사례 단원을 참조하십시오.

권한을 부여할 때 권한을 받는 사용자, 받는 권한의 대상이 되는 리소스, 받는 권한으로 수행할 수 있는 작업을 결정합니다.

주제

- CloudFront 리소스의 ARN (p. 430)
- 리소스 소유권 이해 (p. 430)
- 리소스 액세스 관리 (p. 430)
- 정책 요소 지정: 리소스, 작업, 효과, 보안 주체 (p. 433)
- 정책에서 조건 지정 (p. 434)

CloudFront 리소스의 ARN

웹 및 RTMP 배포, 무효화, 원본 액세스 ID 등의 모든 CloudFront 리소스는 Amazon 리소스 이름(ARN)에 동일한 형식을 사용합니다.

```
arn:aws:cloudfront::optional-account-id:*
```

CloudFront는 이러한 각 유형의 리소스에서 작업하기 위한 API 작업을 제공합니다. 자세한 내용은 Amazon CloudFront API Reference단원을 참조하십시오. 작업과 각 작업을 사용할 권한을 부여하거나 거부하기 위해 지정하는 ARN의 목록은 CloudFront API 권한: 작업, 리소스 및 조건 참조 (p. 439) 단원을 참조하십시오.

리소스 소유권 이해

AWS 계정은 리소스를 누가 생성했든 상관없이 계정에서 생성된 리소스를 소유합니다. 특히, 리소스 소유자는 리소스 생성 요청을 인증하는 보안 주체 엔터티(즉, 루트 계정, IAM 사용자 또는 IAM 역할)의 AWS 계정입니다.

다음 예에서는 이 계정의 작동 방식을 설명합니다.

- AWS 계정의 루트 계정 자격 증명을 사용하여 웹 배포를 생성하는 경우 AWS 계정이 배포의 소유자입니다.
- AWS 계정에서 IAM 사용자를 생성하고 해당 사용자에게 웹 배포를 생성할 수 있는 권한을 부여하는 경우 이 사용자가 웹 배포를 생성할 수 있습니다. 사용자를 생성한 AWS 계정이 배포를 소유합니다.
- AWS 계정에서 웹 배포를 생성할 권한이 있는 IAM 역할을 생성하는 경우 해당 역할을 담당할 수 있는 사람은 누구나 웹 배포를 생성할 수 있습니다. 이 경우 역할이 속한 AWS 계정이 배포를 소유합니다.

리소스 액세스 관리

권한 정책은 누가 무엇에 액세스 할 수 있는지를 지정합니다. 이 단원에서는 CloudFront에 대한 권한 정책을 생성하기 위한 옵션을 설명합니다. IAM 정책 구문과 설명에 대한 일반적인 내용은 IAM 사용 설명서의 AWS IAM 정책 참조를 참조하십시오.

IAM 자격 증명에 연결된 정책을 자격 증명 기반 정책(IAM 정책)이라고 하고, 리소스에 연결된 정책을 리소스 기반 정책이라고 합니다. CloudFront는 리소스 기반 정책을 지원하지 않지만 리소스 수준 정책을 지원합니

다. 리소스 액세스 관리 (p. 430)가 지원하는 권한 정책 유형에 대한 자세한 내용은 CloudFront 단원을 참조하십시오.

주제

- 자격 증명 기반 정책(IAM 정책) (p. 431)
- 리소스 수준 정책 (p. 431)
- 태그 기반 정책 (p. 432)

자격 증명 기반 정책(IAM 정책)

정책을 IAM 자격 증명에 연결할 수 있습니다. 예를 들면,

- 계정 내 사용자 또는 그룹에 권한 정책 연결 –계정 관리자는 특정 사용자에 연결된 권한 정책을 사용하여 해당 사용자에게 웹 배포 만들기 권한을 부여할 수 있습니다.
- 역할에 권한 정책 연결(교차 계정 권한 부여) – 다른 AWS 계정에서 생성된 사용자에게 CloudFront 작업을 수행할 권한을 부여할 수 있습니다. 이렇게 하려면 권한 정책을 IAM 역할에 연결한 다음 다른 계정의 사용자가 역할을 담당할 수 있도록 허용합니다. 다음 예제에서는 계정 A와 계정 B라는 두 개의 AWS 계정에 대해 이 작업을 적용하는 방법을 설명합니다.
  1. 계정 A 관리자는 IAM 역할을 생성하고 계정 A가 소유한 리소스를 생성하거나 액세스할 권한을 부여하는 권한 정책을 역할에 연결합니다.
  2. 계정 A 관리자는 신뢰 정책을 역할에 연결합니다. 신뢰 정책은 역할을 담당할 수 있는 보안 주체로 계정 B를 식별합니다.
  3. 그런 다음 계정 B 관리자는 역할을 담당할 권한을 계정 B의 사용자 또는 그룹에게 위임할 수 있습니다. 이렇게 하면 계정 B의 사용자가 계정 A에서 리소스를 생성하거나 액세스할 수 있습니다.

  다른 AWS 계정의 사용자에게 권한을 위임하는 방법에 대한 자세한 내용은 IAM 사용 설명서의 액세스 관리를 참조하십시오.

다음 예제 정책은 사용자가 `CreateDistribution` 작업을 수행하여 AWS 계정에 대한 웹 배포를 프로그래밍 방식으로 생성할 수 있도록 허용합니다.

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "cloudfront:CreateDistribution"
         ],
         "Resource":"*"
      }
   ]
}
```

CloudFront 콘솔을 사용하여 작업을 수행하는 데 필요한 권한에 대한 자세한 내용은 CloudFront 콘솔 사용에 필요한 권한 (p. 435) 단원을 참조하십시오. CloudFront 자격 증명에 정책 연결에 대한 자세한 내용은 CloudFront에 대한 자격 증명 기반 정책(IAM 정책) 사용 (p. 434) 단원을 참조하십시오. 사용자, 그룹, 역할 및 권한에 대한 자세한 내용은 IAM 사용 설명서의 자격 증명(사용자, 그룹 및 역할)을 참조하십시오.

리소스 수준 정책

일부 시나리오에서는 지정한 리소스에 대한 특정 수준의 액세스 권한을 부여할 수 있습니다. 예를 들어, 해당 리소스에 대한 특정 작업에 대한 액세스만 부여할 수 있습니다. 일부 AWS 서비스에서 이 정책을 구현하는 한 가지 방법은 리소스에 대한 정책을 직접 연결하도록 허용하는 것입니다. 예를 들면, Amazon S3와 Elasticsearch에서 리소스 액세스 제어를 구현하는 방법이 있습니다. CloudFront는 동일한 유연성을 허용하지만 다른 방법을 사용합니다. 정책을 리소스에 연결하는 대신, 정책에서 리소스를 지정합니다.

예를 들어, 다음 정책은 배포의 ARN별로 지정한 배포에 대한 업데이트, 삭제 및 생성 무효화 액세스를 허용할 수 있는 방법을 보여 줍니다. 이 정책은 AWS API 또는 AWS CLI에서만 이러한 작업을 완료하는 데 필요한 권한을 부여합니다. (이 정책을 사용하려면 정책 예제의 기울임꼴 텍스트를 자신의 리소스 정보로 바꾸십시오.)

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "cloudfront:CreateDistribution",
                "cloudfront:Get*",
                "cloudfront:List*"
            ],
            "Resource": "*"
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": [
                "cloudfront:UpdateDistribution",
                "cloudfront:DeleteDistribution",
                "cloudfront:CreateInvalidation",

            ],
            "Resource": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE"
        }
    ]
}
```

태그 기반 정책

IAM 정책을 설계할 때 특정 리소스에 대한 액세스 권한을 부여하여 세부적인 권한을 설정할 수 있습니다. 관리하는 리소스의 개수가 늘어날수록 이 작업은 더 어려워집니다. 리소스에 태그를 지정하고 정책 문 조건에서 태그를 사용하면 이러한 작업이 더 간단해질 수 있습니다. 특정 태그를 사용하여 리소스에 대량으로 액세스 권한을 부여합니다. 그런 다음 생성 중 또는 나중에 이 태그를 관련 리소스에 반복해서 적용합니다.

> **Note**
>
> 조건에 태그를 사용하는 것은 리소스 및 요청에 대한 액세스를 제어하는 하나의 방법입니다. CloudFront의 태그 지정에 대한 자세한 내용은 Amazon CloudFront 배포에 태그 지정 (p. 424) 단원을 참조하십시오.

리소스에 태그가 연결되거나 태그 지정을 지원하는 서비스에 대한 요청에서 전달될 수 있습니다. CloudFront에서는 리소스에 태그가 있을 수 있으며 일부 작업에 태그가 포함될 수 있습니다. IAM 정책을 생성할 때 태그 조건 키를 사용하여 다음을 제어할 수 있습니다.

- 이미 가지고 있는 태그를 기반으로 어떤 사용자가 배포에 대해 작업을 수행할 수 있는지를 제어합니다.
- 어떤 태그가 작업의 요청에서 전달될 수 있는지 제어합니다.
- 요청에서 특정 키를 사용할 수 있는지 여부를 제어합니다.

태그 조건 키의 전체 구문 및 의미는 IAM 사용 설명서의 IAM 태그를 사용한 액세스 제어를 참조하십시오.

예를 들어, `AWSCloudFrontFullAccess` 관리형 정책은 사용자에게 모든 리소스에서 모든 CloudFront 작업을 수행할 수 있는 무제한적인 권한을 부여합니다. 다음 정책은 이러한 기능을 제한하고 권한이 없는 사용자가 CloudFront 프로덕션 배포를 생성할 수 있는 권한을 거부합니다.

태그를 사용하여 제한을 구현하려면 정책은 요청이 `gamma` 또는 `prod` 값 중 하나를 사용하여 `stage`라는 태그를 지정하는 경우 `CreateDistribution` 작업을 거부합니다. 또한 정책은 이러한 동일한 태그 값을 포함하거나 `stage` 태그를 완전하게 제거하기 위한 태그 수정 작업을 허용하지 않음으로써 이러한 권한이 없는 사용자가 프로덕션 환경의 단계를 변경하지 못하도록 합니다. 고객의 관리자는 관리형 사용자 정책 이외에 이 IAM 정책도 권한이 없는 IAM 사용자에게 연결해야 합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "cloudfront:CreateDistribution",
                "cloudfront:TagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/stage": [
                        "gamma",
                        "prod"
                    ]
                }
            }
        },
        {
            "Effect": "Deny",
            "Action": [
                "cloudfront:TagResource",
                "cloudfront:UntagResource"
            ],
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": [
                        "stage"
                    ]
                },
                "StringEquals": {
                    "aws:ResourceTag/stage": [
                        "gamma",
                        "prod"
                    ]
                }
            }
        }
    ]
}
```

정책 요소 지정: 리소스, 작업, 효과, 보안 주체

CloudFront에는 각 CloudFront 리소스에서 사용할 수 있는 API 작업(Amazon CloudFront API Reference 참조)이 포함됩니다(CloudFront 리소스의 ARN (p. 430) 참조). 사용자 또는 연합된 사용자에게 이러한 작업 중 하나 또는 전부를 수행할 권한을 부여할 수 있습니다.

다음은 기본 정책 요소입니다.

- 리소스 – Amazon 리소스 이름(ARN)을 사용하여 정책을 적용할 리소스를 식별합니다. 자세한 내용은 CloudFront 리소스의 ARN (p. 430) 단원을 참조하십시오.
- 작업 – 작업 키워드를 사용하여 허용 또는 거부할 리소스 작업을 식별합니다. 예를 들어, 지정된 `Effect`에 따라 `cloudfront:CreateDistribution` 권한은 CloudFront `CreateDistribution` 작업을 수행할 사용자 권한을 허용하거나 거부합니다.

- Effect – 사용자가 지정된 리소스에서 작업을 수행하려고 할 때 효과를 허용 또는 거부로 지정합니다. 작업에 대한 액세스 권한을 명시적으로 부여하지 않으면 액세스는 묵시적으로 거부됩니다. 다른 정책에서 액세스 권한을 부여하는 경우라도 사용자가 해당 리소스에 액세스할 수 없도록 하기 위해 리소스에 대한 권한을 명시적으로 거부할 수도 있습니다.
- 보안 주체 – 자격 증명 기반 정책(IAM 정책)에서 정책이 연결되는 사용자는 암시적인 보안 주체입니다. 리소스 기반 정책의 경우 사용자, 계정, 서비스 또는 권한의 수신자인 기타 엔터티를 지정합니다(리소스 기반 정책에만 해당).

  CloudFront는 리소스 기반 정책을 지원하지 않지만 리소스 수준 정책을 지원합니다. 리소스 액세스 관리 (p. 430)가 지원하는 권한 정책 유형에 대한 자세한 내용은 CloudFront 단원을 참조하십시오.

IAM 정책 구문과 설명에 대한 자세한 내용은 IAM 사용 설명서의 AWS IAM 정책 참조를 참조하십시오.

모든 CloudFront API 작업과 해당 작업이 적용되는 리소스를 보여주는 목록은 CloudFront API 권한: 작업, 리소스 및 조건 참조 (p. 439) 단원을 참조하십시오.

정책에서 조건 지정

권한을 부여할 때 IAM 정책 언어를 사용하여 정책이 언제 적용되는지를 지정할 수 있습니다. 예를 들어, 특정 날짜 이후에만 정책을 적용할 수 있습니다. 정책 언어에서의 조건 지정에 관한 자세한 내용은 IAM 사용 설명서의 조건 단원을 참조하십시오.

조건을 표시하려면 미리 정의된 조건 키를 사용합니다. CloudFront에만 해당되는 특정한 조건 키는 없습니다. 하지만 적절하게 사용할 수 있는 AWS 차원의 조건 키가 있습니다. AWS 차원 키의 전체 목록은 IAM 사용 설명서의 사용 가능한 조건 키 단원을 참조하십시오.

CloudFront에 대한 자격 증명 기반 정책(IAM 정책) 사용

이 주제에서는 자격 증명 기반 정책의 예를 통해 계정 관리자가 IAM 자격 증명(사용자, 그룹, 역할)에 권한 정책을 연결함으로써 CloudFront 리소스에 대한 작업 수행 권한을 부여하는 방법을 보여 줍니다.

> **Important**
>
> CloudFront 리소스 액세스를 관리하기 위한 기본 개념과 옵션을 설명하는 개요 주제를 먼저 읽어 보는 것이 좋습니다. 자세한 내용은 CloudFront 리소스에 대한 액세스 권한 관리 개요 (p. 430) 단원을 참조하십시오.

주제

- CloudFront 콘솔 사용에 필요한 권한 (p. 435)
- CloudFront에 대한 AWS 관리형(미리 정의된) 정책 (p. 436)
- 고객 관리형 정책 예 (p. 437)

다음에서는 권한 정책을 보여 줍니다. `Sid`(명령문 ID)는 선택 사항입니다.

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowAllCloudFrontPermissions",
         "Effect": "Allow",
         "Action": ["cloudfront:*"],
         "Resource": "*"
      }
   ]
}
```

정책은 모든 CloudFront 작업을 수행할 수 있는 권한을 부여하며, 이러한 권한은 프로그래밍 방식으로 CloudFront에 액세스하는 데 충분합니다. 콘솔을 사용하여 CloudFront에 액세스하는 경우 CloudFront 콘솔 사용에 필요한 권한 (p. 435) 단원을 참조하십시오.

작업과 각 작업을 사용할 권한을 부여하거나 거부하기 위해 지정하는 ARN의 목록은 CloudFront API 권한: 작업, 리소스 및 조건 참조 (p. 439) 단원을 참조하십시오.

CloudFront 콘솔 사용에 필요한 권한

CloudFront 콘솔에 대한 전체 액세스 권한을 부여하려면 다음 권한 정책에서 권한을 부여합니다.

```
{
   "Version": "2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "acm:ListCertificates",
            "cloudfront:*",
            "cloudwatch:DescribeAlarms",
            "cloudwatch:PutMetricAlarm",
            "cloudwatch:GetMetricStatistics",
            "elasticloadbalancing:DescribeLoadBalancers",
            "iam:ListServerCertificates",
            "sns:ListSubscriptionsByTopic",
            "sns:ListTopics",
            "waf:GetWebACL",
            "waf:ListWebACLs"
         ],
         "Resource":"*"
      },
      {
         "Effect":"Allow",
         "Action":[
            "s3:ListAllMyBuckets",
            "s3:PutBucketPolicy"
         ],
         "Resource":"arn:aws:s3:::*"
      }
   ]
}
```

다음은 권한이 필요한 이유입니다.

`acm:ListCertificates`

CloudFront 콘솔을 사용하여 웹 배포를 생성하고 업데이트할 때 최종 사용자와 CloudFront 간에 또는 CloudFront와 오리진 간에 HTTPS를 요구하도록 CloudFront를 구성하려는 경우, ACM 인증서 목록을 볼 수 있도록 허용합니다.

CloudFront 콘솔을 사용하지 않는 경우에는 이 권한이 필요하지 않습니다.

`cloudfront:*`

모든 CloudFront 작업을 수행할 수 있도록 허용합니다.

`cloudwatch:DescribeAlarms` 및 `cloudwatch:PutMetricAlarm`

CloudFront 콘솔에서 CloudWatch 경보를 생성하고 볼 수 있습니다. `sns:ListSubscriptionsByTopic` 및 `sns:ListTopics`도 참조하십시오.

CloudFront 콘솔을 사용하지 않는 경우에는 이러한 권한이 필요하지 않습니다.

`cloudwatch:GetMetricStatistics`

CloudFront 콘솔에서 CloudFront가 CloudWatch 측정치를 렌더링할 수 있도록 허용합니다.

CloudFront 콘솔을 사용하지 않는 경우에는 이 권한이 필요하지 않습니다.

`elasticloadbalancing:DescribeLoadBalancers`

웹 배포를 생성하고 업데이트할 때, 사용 가능한 원본 목록에서 Elastic Load Balancing 로드 밸런서를 볼 수 있도록 허용합니다.

CloudFront 콘솔을 사용하지 않는 경우에는 이 권한이 필요하지 않습니다.

`iam:ListServerCertificates`

CloudFront 콘솔을 사용하여 웹 배포를 생성하고 업데이트할 때 최종 사용자와 CloudFront 사이 또는 CloudFront와 오리진 간에 HTTPS를 요구하도록 CloudFront를 구성하려는 경우, IAM 인증서 저장소에서 인증서 목록을 볼 수 있도록 허용합니다.

CloudFront 콘솔을 사용하지 않는 경우에는 이 권한이 필요하지 않습니다.

`s3:ListAllMyBuckets`

웹 및 RTMP 배포를 생성하고 업데이트할 때, 다음 작업을 수행할 수 있도록 허용합니다.

- 사용 가능한 원본 목록에서 S3 버킷 목록 보기
- 액세스 로그를 저장할 수 있는 S3 버킷 목록 보기

CloudFront 콘솔을 사용하지 않는 경우에는 이 권한이 필요하지 않습니다.

`S3:PutBucketPolicy`

액세스를 S3 버킷으로 제한하는 배포를 생성하거나 업데이트할 때, 사용자가 버킷 정책을 업데이트하여 CloudFront 원본 액세스 ID에 대한 액세스 권한을 부여할 수 있도록 허용합니다. 자세한 내용은 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한 (p. 176) 단원을 참조하십시오.

CloudFront 콘솔을 사용하지 않는 경우에는 이 권한이 필요하지 않습니다.

`sns:ListSubscriptionsByTopic` 및 `sns:ListTopics`

CloudFront 콘솔에서 CloudWatch 경보를 생성할 때, 알림에 대한 SNS 주제를 선택할 수 있도록 허용합니다.

CloudFront 콘솔을 사용하지 않는 경우에는 이러한 권한이 필요하지 않습니다.

`waf:GetWebACL` 및 `waf:ListWebACLs`

CloudFront 콘솔에서 AWS WAF 웹 ACL 목록을 볼 수 있도록 허용합니다.

CloudFront 콘솔을 사용하지 않는 경우에는 이러한 권한이 필요하지 않습니다.

CloudFront에 대한 AWS 관리형(미리 정의된) 정책

AWS는 AWS에서 생성하고 관리하는 독립형 IAM 정책을 제공하여 많은 일반 사용 사례를 처리합니다. 이러한 AWS 관리형 정책은 사용자가 필요한 권한을 조사할 필요가 없도록 일반 사용 사례에 필요한 권한을 부여합니다. 자세한 내용은 IAM 사용 설명서의 AWS 관리형 정책 단원을 참조하십시오. CloudFront의 경우 IAM은 다음 두 가지 관리형 정책을 제공합니다.

- CloudFrontFullAccess – CloudFront 리소스에 대한 전체 액세스 권한을 부여합니다.

  > **Important**
  >
  > CloudFront에서 액세스 로그를 생성하여 저장하도록 하려면 추가 권한을 부여해야 합니다. 자세한 내용은 로그 기록 및 로그 파일에 대한 액세스를 구성하는 데 필요한 권한 (p. 395) 단원을 참조하십시오.

- CloudFrontReadOnlyAccess – CloudFront 리소스에 대한 읽기 전용 액세스 권한을 부여합니다.

Note

IAM 콘솔에 로그인하고 이 콘솔에서 특정 정책을 검색하여 이러한 권한 정책을 검토할 수 있습니다. CloudFront API 작업에 대한 권한을 허용하는 고유의 사용자 지정 IAM 정책을 생성할 수도 있습니다. 해당 권한이 필요한 IAM 사용자 또는 그룹에 이러한 사용자 지정 정책을 연결할 수 있습니다.

고객 관리형 정책 예

CloudFront API 작업에 대한 권한을 허용하는 고유의 사용자 지정 IAM 정책을 생성할 수 있습니다. 지정된 권한이 필요한 IAM 사용자 또는 그룹에 이러한 사용자 지정 정책을 연결할 수 있습니다. 이러한 정책은 CloudFront API, AWS SDK 또는 AWS CLI를 사용하는 경우에 유효합니다. 다음 예제에서는 몇 가지 일반 사용 사례에 대한 권한을 보여 줍니다. 사용자에게 CloudFront에 대한 전체 액세스 권한을 부여하는 정책에 대해서는 CloudFront 콘솔 사용에 필요한 권한 (p. 435) 단원을 참조하십시오.

예제

- 예제 1: 모든 웹 배포에 대한 읽기 액세스 허용 (p. 437)
- 예제 2: 웹 배포의 생성, 업데이트 및 삭제 허용 (p. 437)
- 예제 3: 무효화 생성 및 나열 허용 (p. 438)

예제 1: 모든 웹 배포에 대한 읽기 액세스 허용

다음 권한 정책은 사용자에게 CloudFront 콘솔에서 모든 웹 배포를 볼 수 있는 사용자 권한을 부여합니다.

```
{
   "Version": "2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "acm:ListCertificates",
            "cloudfront:GetDistribution",
            "cloudfront:GetDistributionConfig",
            "cloudfront:ListDistributions",
            "cloudfront:ListCloudFrontOriginAccessIdentities",
            "elasticloadbalancing:DescribeLoadBalancers",
            "iam:ListServerCertificates",
            "sns:ListSubscriptionsByTopic",
            "sns:ListTopics",
            "waf:GetWebACL",
            "waf:ListWebACLs"
         ],
         "Resource":"*"
      },
      {
         "Effect":"Allow",
         "Action":[
            "s3:ListAllMyBuckets"
         ],
         "Resource":"arn:aws:s3:::*"
      }
   ]
}
```

예제 2: 웹 배포의 생성, 업데이트 및 삭제 허용

다음 권한 정책은 사용자가 CloudFront 콘솔을 사용하여 웹 배포를 생성, 업데이트 및 삭제할 수 있도록 허용합니다.

```
{
```

```
   "Version": "2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "acm:ListCertificates",
            "cloudfront:CreateDistribution",
            "cloudfront:DeleteDistribution",
            "cloudfront:GetDistribution",
            "cloudfront:GetDistributionConfig",
            "cloudfront:ListDistributions",
            "cloudfront:UpdateDistribution",
            "cloudfront:ListCloudFrontOriginAccessIdentities",
            "elasticloadbalancing:DescribeLoadBalancers",
            "iam:ListServerCertificates",
            "sns:ListSubscriptionsByTopic",
            "sns:ListTopics",
            "waf:GetWebACL",
            "waf:ListWebACLs"
         ],
         "Resource":"*"
      },
      {
         "Effect":"Allow",
         "Action":[
            "s3:ListAllMyBuckets",
            "s3:PutBucketPolicy"
         ],
         "Resource":"arn:aws:s3:::*"
      }
   ]
}
```

`cloudfront:ListCloudFrontOriginAccessIdentities` 권한이 있으면 사용자가 Amazon S3 버킷의 객체에 액세스할 권한을 기존 원본 액세스 ID에 자동으로 부여할 수 있습니다. 또한 사용자가 원본 액세스 ID를 생성할 수 있도록 하려면 `cloudfront:CreateCloudFrontOriginAccessIdentity` 권한도 허용해야 합니다.

예제 3: 무효화 생성 및 나열 허용

다음 권한 정책은 사용자가 무효화를 생성하고 나열할 수 있도록 허용합니다. 먼저 배포에 대한 설정을 표시하여 무효화를 생성하고 보기 때문에 CloudFront 배포에 대한 읽기 액세스 권한도 여기에 포함됩니다.

```
{
   "Version": "2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "acm:ListCertificates",
            "cloudfront:GetDistribution",
            "cloudfront:GetStreamingDistribution",
            "cloudfront:GetDistributionConfig",
            "cloudfront:ListDistributions",
            "cloudfront:ListCloudFrontOriginAccessIdentities",
            "cloudfront:CreateInvalidation",
            "cloudfront:GetInvalidation",
            "cloudfront:ListInvalidations",
            "elasticloadbalancing:DescribeLoadBalancers",
            "iam:ListServerCertificates",
            "sns:ListSubscriptionsByTopic",
            "sns:ListTopics",
            "waf:GetWebACL",
            "waf:ListWebACLs"
```

```
            ],
            "Resource":"*"
        },
        {
            "Effect":"Allow",
            "Action":[
                "s3:ListAllMyBuckets"
            ],
            "Resource":"arn:aws:s3:::*"
        }
    ]
}
```

CloudFront API 권한: 작업, 리소스 및 조건 참조

액세스 제어 (p. 429) 설정 및 IAM 자격 증명에 연결할 수 있는 권한 정책(자격 증명 기반 정책) 작성 시 다음 목록을 참조로 사용할 수 있습니다. 에는 CloudFront API 작업, 수행하도록 권한을 부여할 수 있는 작업, 권한을 부여할 수 있는 AWS 리소스가 각각 나열되어 있습니다. 정책의 `Action` 필드에서 작업을 지정하고, 정책의 `Resource` 필드에서 리소스 값을 지정합니다.

CloudFront 정책에서 AWS 차원 조건 키를 사용하여 조건을 표시할 수 있습니다. AWS 차원 키의 전체 목록은 IAM 사용 설명서의 사용할 수 있는 키 단원을 참조하십시오.

주제

- 웹 배포 작업에 필요한 권한 (p. 439)
- RTMP 배포 작업에 필요한 권한 (p. 440)
- 무효화 작업에 필요한 권한 (p. 442)
- 원본 액세스 ID 작업에 필요한 권한 (p. 442)
- Lambda@Edge 관련 CloudFront 작업에 필요한 권한 (p. 442)
- 태그 작업에 필요한 권한 (p. 443)

웹 배포 작업에 필요한 권한

CreateDistribution

필요한 권한(API 작업):

- `cloudfront:CreateDistribution`
- `acm:ListCertificates`(CloudFront 콘솔 전용)
- CloudFront에서 액세스 로그를 저장하도록 구성할 경우에만:
  - `s3:GetBucketAcl`
  - `s3:PutBucketAcl`
  - 버킷의 S3 ACL에서 `FULL_CONTROL` 권한을 부여받아야 합니다.

리소스:

- CloudFront: *
- ACM: *
- Amazon S3: 액세스 로그를 저장하도록 CloudFront를 구성할 경우 특정 버킷에 대한 액세스를 선택적으로 제한할 수 있습니다.

CreateDistributionWithTags

필요한 권한(API 작업):

- `cloudfront:CreateDistribution`, `cloudfront:TagResource`
- `acm:ListCertificates`(CloudFront 콘솔 전용)

- CloudFront에서 액세스 로그를 저장하도록 구성할 경우에만:
  - `s3:GetBucketAcl`
  - `s3:PutBucketAcl`
  - 버킷의 S3 ACL에서 `FULL_CONTROL` 권한을 부여받아야 합니다.

리소스:

- CloudFront: *
- ACM: *
- Amazon S3: 액세스 로그를 저장하도록 CloudFront를 구성할 경우 특정 버킷에 대한 액세스를 선택적으로 제한할 수 있습니다.

GetDistribution

필요한 권한(API 작업): `cloudfront:GetDistribution`, `acm:ListCertificates` (CloudFront 콘솔 전용)

리소스: `*`

GetDistributionConfig

필요한 권한(API 작업): `cloudfront:GetDistributionConfig`, `acm:ListCertificates` (CloudFront 콘솔 전용)

리소스: `*`

ListDistributions

필요한 권한(API 작업): `cloudfront:ListDistributions`

리소스: `*`

UpdateDistribution

필요한 권한(API 작업):

- `cloudfront:UpdateDistribution`
- `acm:ListCertificates`(CloudFront 콘솔 전용)
- CloudFront에서 액세스 로그를 저장하도록 구성할 경우에만:
  - `s3:GetBucketAcl`
  - `s3:PutBucketAcl`
  - 버킷의 S3 ACL에서 `FULL_CONTROL` 권한을 부여받아야 합니다.

리소스:

- CloudFront: *
- ACM: *
- Amazon S3: 액세스 로그를 저장하도록 CloudFront를 구성할 경우 특정 버킷에 대한 액세스를 선택적으로 제한할 수 있습니다.

DeleteDistribution

필요한 권한(API 작업): `cloudfront:DeleteDistribution`

리소스: `*`

RTMP 배포 작업에 필요한 권한

CreateStreamingDistribution

필요한 권한(API 작업): `cloudfront:CreateStreamingDistribution`

CloudFront에서 액세스 로그를 저장하도록 구성할 경우에만:

- `s3:GetBucketAcl`
- `s3:PutBucketAcl`
- 버킷의 S3 ACL에서 `FULL_CONTROL` 권한을 부여받아야 합니다.

리소스: `*`

CloudFront에서 액세스 로그를 저장하도록 구성할 경우 특정 버킷에 대한 액세스를 선택적으로 제한할 수 있습니다.

CreateStreamingDistributionWithTags

필요한 권한(API 작업): `cloudfront:CreateStreamingDistribution`, `cloudfront:TagResource`

CloudFront에서 액세스 로그를 저장하도록 구성할 경우에만:

- `s3:GetBucketAcl`
- `s3:PutBucketAcl`
- 버킷의 S3 ACL에서 `FULL_CONTROL` 권한을 부여받아야 합니다.

리소스: `*`

CloudFront에서 액세스 로그를 저장하도록 구성할 경우 특정 버킷에 대한 액세스를 선택적으로 제한할 수 있습니다.

GetStreamingDistribution

필요한 권한(API 작업): `cloudfront:GetStreamingDistribution`

리소스: `*`

GetStreamingDistributionConfig

필요한 권한(API 작업): `cloudfront:GetStreamingDistributionConfig`

리소스: `*`

ListStreamingDistributions

필요한 권한(API 작업): `cloudfront:ListStreamingDistributions`

리소스: `*`

UpdateStreamingDistribution

필요한 권한(API 작업): `cloudfront:UpdateStreamingDistribution`

CloudFront에서 액세스 로그를 저장하도록 구성할 경우에만:

- `s3:GetBucketAcl`
- `s3:PutBucketAcl`
- 버킷의 S3 ACL에서 `FULL_CONTROL` 권한을 부여받아야 합니다.

리소스: `*`

CloudFront에서 액세스 로그를 저장하도록 구성할 경우 특정 버킷에 대한 액세스를 선택적으로 제한할 수 있습니다.

DeleteStreamingDistribution

필요한 권한(API 작업): `cloudfront:DeleteDistribution`

리소스: `*`

무효화 작업에 필요한 권한

CreateInvalidation

필요한 권한(API 작업): `cloudfront:CreateInvalidation`

리소스: `*`

GetInvalidation

필요한 권한(API 작업): `cloudfront:GetInvalidation`

리소스: `*`

ListInvalidations

필요한 권한(API 작업): `cloudfront:ListInvalidations`

리소스: `*`

원본 액세스 ID 작업에 필요한 권한

CreateCloudFrontOriginAccessIdentity

필요한 권한(API 작업): `cloudfront:CreateCloudFrontOriginAccessIdentity`

리소스: `*`

GetCloudFrontOriginAccessIdentity

필요한 권한(API 작업): `cloudfront:GetCloudFrontOriginAccessIdentity`

리소스: `*`

GetCloudFrontOriginAccessIdentityConfig

필요한 권한(API 작업): `cloudfront:GetCloudFrontOriginAccessIdentityConfig`

리소스: `*`

ListCloudFrontOriginAccessIdentities

필요한 권한(API 작업): `cloudfront:ListDistributions`

리소스: `*`

UpdateCloudFrontOriginAccessIdentity

필요한 권한(API 작업): `cloudfront:UpdateCloudFrontOriginAccessIdentity`

리소스: `*`

DeleteCloudFrontOriginAccessIdentity

필요한 권한(API 작업): `cloudfront:DeleteCloudFrontOriginAccessIdentity`

리소스: `*`

Lambda@Edge 관련 CloudFront 작업에 필요한 권한

Lambda@Edge를 사용하려면 Lambda 함수 트리거를 포함하는 배포를 생성하거나 업데이트할 수 있도록 다음의 CloudFront 권한이 필요합니다. 필요한 Lambda 권한에 대한 자세한 내용은 AWS Lambda Developer Guide의 "AWS Lambda@Edge" 장에서 IAM 권한 설정을 참조하십시오.

CreateDistribution

필요한 권한(API 작업):

- `cloudfront:CreateDistribution`
- `acm:ListCertificates`(CloudFront 콘솔 전용)
- CloudFront에서 액세스 로그를 저장하도록 구성할 경우에만:
  - `s3:GetBucketAcl`
  - `s3:PutBucketAcl`
  - 버킷의 S3 ACL에서 `FULL_CONTROL` 권한을 부여받아야 합니다.

리소스:

- CloudFront: *
- ACM: *
- Amazon S3: 액세스 로그를 저장하도록 CloudFront를 구성할 경우 특정 버킷에 대한 액세스를 선택적으로 제한할 수 있습니다.

CreateDistributionWithTags

필요한 권한(API 작업):

- `cloudfront:CreateDistribution`, `cloudfront:TagResource`
- `acm:ListCertificates`(CloudFront 콘솔 전용)
- CloudFront에서 액세스 로그를 저장하도록 구성할 경우에만:
  - `s3:GetBucketAcl`
  - `s3:PutBucketAcl`
  - 버킷의 S3 ACL에서 `FULL_CONTROL` 권한을 부여받아야 합니다.

리소스:

- CloudFront: *
- ACM: *
- Amazon S3: 액세스 로그를 저장하도록 CloudFront를 구성할 경우 특정 버킷에 대한 액세스를 선택적으로 제한할 수 있습니다.

UpdateDistribution

필요한 권한(API 작업):

- `cloudfront:UpdateDistribution`
- `acm:ListCertificates`(CloudFront 콘솔 전용)
- CloudFront에서 액세스 로그를 저장하도록 구성할 경우에만:
  - `s3:GetBucketAcl`
  - `s3:PutBucketAcl`
  - 버킷의 S3 ACL에서 `FULL_CONTROL` 권한을 부여받아야 합니다.

리소스:

- CloudFront: *
- ACM: *
- Amazon S3: 액세스 로그를 저장하도록 CloudFront를 구성할 경우 특정 버킷에 대한 액세스를 선택적으로 제한할 수 있습니다.

태그 작업에 필요한 권한

TagResource

필요한 권한(API 작업): `cloudfront:TagResource`

리소스: *

UntagResource

필요한 권한(API 작업): `cloudfront:UntagResource`

리소스: *

ListTagsForResource

필요한 권한(API 작업): `cloudfront:ListTagsForResource`

리소스: *

Amazon CloudFront의 로깅 및 모니터링

모니터링은 CloudFront와 사용자 AWS 솔루션의 가용성 및 성능을 유지하는 중요한 역할을 합니다. 다중 지점 실패가 발생할 경우 보다 쉽게 디버깅할 수 있도록 AWS 솔루션의 모든 부분으로부터 모니터링 데이터를 수집해야 합니다. AWS는 CloudFront 리소스와 활동을 모니터링하고 잠재적 인시던트에 대응하기 위한 여러 도구를 제공합니다.

Amazon CloudWatch 경보

CloudWatch 경보를 사용하면 지정한 기간 동안 단일 지표를 감시합니다. 지표가 지정된 임계값을 초과하면 Amazon SNS 주제 또는 AWS Auto Scaling 정책에 알림이 전송됩니다. CloudWatch 경보는 지표가 특정 상태에 있는 경우 작업을 호출하지 않습니다. 대신, 상태가 변경되어 지정된 기간 동안 유지되어야 합니다. 자세한 내용은 Amazon CloudWatch를 사용하여 CloudFront 모니터링 (p. 416) 단원을 참조하십시오.

AWS CloudTrail 로그

CloudTrail은 CloudFront에서 사용자, 역할 또는 AWS 제품이 수행한 작업의 레코드를 제공합니다. CloudTrail에서 수집하는 정보를 사용하여 CloudFront에 수행된 요청, 요청이 수행된 IP 주소, 요청을 수행한 사람, 요청이 수행된 시간 및 추가 세부 정보를 확인할 수 있습니다. 자세한 내용은 AWS CloudTrail를 사용하여 CloudFront API로 전송된 요청 캡처 (p. 410) 단원을 참조하십시오.

CloudFront 액세스 로그

서버 액세스 로그는 배포에 대해 이루어진 요청에 대한 상세 레코드를 제공합니다. 서버 액세스 로그는 많은 애플리케이션에 있어 유용합니다. 예를 들어 액세스 로그 정보는 보안 및 액세스 감사에 유용할 수 있습니다. 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.

CloudFront 콘솔 보고서

CloudFront 콘솔에는 캐시 통계 보고서, 많이 사용되는 객체 보고서 및 주요 참조 보고서 등의 다양한 보고서가 포함되어 있습니다. 대부분의 CloudFront 콘솔 보고서는 CloudFront 액세스 로그의 데이터를 기반으로 합니다. 이 로그에는 CloudFront가 수신하는 모든 사용자 요청에 대한 세부 정보가 포함되어 있습니다. 그러나 보고서를 보기 위해 액세스 로그를 활성화할 필요는 없습니다. 자세한 내용은 콘솔의 CloudFront 보고서 (p. 369) 단원을 참조하십시오.

Amazon CloudFront 규정 준수 확인

타사 감사자는 여러 AWS 규정 준수 프로그램의 일환으로 Amazon CloudFront의 보안 및 규정 준수를 평가합니다. 여기에는 SOC, PCI, HIPAA 등이 포함됩니다.

규정 준수 프로그램 범위에 속하는 AWS 서비스의 목록은 규정 준수 프로그램 제공 범위 내 AWS 서비스 페이지에서 확인하십시오. 일반 정보는 AWS 규정 준수 프로그램을 참조하십시오.

AWS Artifact를 사용하여 타사 감사 보고서를 다운로드할 수 있습니다. 자세한 내용은 AWS Artifact의 보고서 다운로드를 참조하십시오.

CloudFront 사용 시 규정 준수 책임은 데이터의 민감도, 회사의 규정 준수 목표 및 관련 법률과 규정에 따라 결정됩니다. AWS에서는 규정 준수를 지원할 다음과 같은 리소스를 제공합니다.

- 보안 및 규정 준수 빠른 시작 안내서 – 이 배포 안내서에서는 아키텍처 고려 사항에 대해 설명하고 보안 및 규정 준수에 중점을 둔 기본 환경을 AWS에 배포하기 위한 단계를 제공합니다.
- HIPAA 보안 및 규정 준수에 맞는 아키텍처 설계 백서 – 이 백서는 회사에서 AWS를 사용하여 HIPAA를 준수하는 애플리케이션을 만드는 방법을 설명합니다.

  AWS HIPAA 규정 준수 프로그램에는 CloudFront가 HIPAA 적격 서비스로 포함되어 있습니다. AWS와 BAA(Business Associate Addendum)를 체결한 경우, CloudFront를 사용하여 PHI(보호 대상 건강 정보)가 포함된 콘텐츠를 제공할 수 있습니다. 자세한 내용은 HIPAA 규정 준수 단원을 참조하십시오.
- AWS 규정 준수 리소스 – 이 워크북 및 안내서는 귀사의 산업 및 위치에 적용될 수 있습니다.
- AWS Config – 이 AWS 서비스로 리소스 구성이 내부 관행, 업계 지침 및 규정을 준수하는 정도를 평가할 수 있습니다.
- AWS Security Hub – 이 AWS 서비스는 보안 업계 표준 및 모범 사례 준수 여부를 확인하는 데 도움이 되는 AWS 내 보안 상태에 대한 포괄적인 관점을 제공합니다.

CloudFront 규정 준수 모범 사례

이 단원에서는 Amazon CloudFront를 사용하여 콘텐츠를 제공할 때 규정 준수를 위해 참고할 수 있는 모범 사례 및 권장 사항을 제공합니다.

AWS 공동 책임 모델에 따라 PCI 또는 HIPAA 규정에 맞는 워크로드를 실행하는 경우, 향후 감사를 위해 과거 365일 간의 CloudFront 사용 데이터를 기록해 두는 것이 좋습니다. 사용 데이터를 기록하는 방법은 다음과 같습니다.

- CloudFront 액세스 로그를 활성화합니다. 자세한 내용은 액세스 로그 구성 및 사용 (p. 393) 단원을 참조하십시오.
- CloudFront API로 전송된 요청을 캡처합니다. 자세한 내용은 AWS CloudTrail를 사용하여 CloudFront API로 전송된 요청 캡처 (p. 410) 단원을 참조하십시오.

또한 CloudFront가 PCI DSS 및 SOC 표준을 준수하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

PCI DSS

CloudFront에서는 전자 상거래 웹사이트 운영자 또는 서비스 공급자에 의한 신용 카드 데이터의 처리, 저장 및 전송을 지원하며, Payment Card Industry(PCI) Data Security Standard(DSS) 준수를 검증 받았습니다. AWS PCI 규정 준수 패키지의 사본을 요청하는 방법 등 PCI DSS에 대해 자세히 알아보려면 PCI DSS 레벨 1을 참조하십시오.

보안을 위해, 신용 카드 정보를 CloudFront 엣지 캐시에 캐싱하지 않는 것이 좋습니다. 예를 들어, 신용 카드 번호 마지막 4자리와 신용카드 소유자의 연락처 정보와 같은 정보를 담고 있는 응답에 `Cache-Control:no-cache="field-name"` 헤더가 포함되도록 오리진을 구성할 수 있습니다.

AWS 시스템 및 조직 제어

CloudFront는 SOC 1, SOC 2 및 SOC 3을 포함한 시스템 및 조직 제어(SOC) 측정값을 준수합니다. SOC 보고서는 AWS가 주요 규정 준수 제어 항목 및 제어 대상을 어떻게 준수하고 있는지를 입증하는 독립적인 타사 심사 보고서입니다. 이러한 감사는 고객 및 회사 데이터의 보안, 기밀성 및 가용성에 영향을 미칠 수 있는 위험으로부터 보호하기 위해 적절한 보호 및 절차가 시행되고 있는지 확인합니다. 이러한 타사 감사의 결과는 AWS SOC 규정 준수 웹 사이트에서 사용할 수 있습니다. 이 사이트에서는 게시된 보고서를 검토하여 AWS 작업 및 규정 준수를 지원하는 제어에 대한 추가 정보를 확인할 수 있습니다.

Amazon CloudFront의 복원성

AWS 글로벌 인프라는 AWS 리전 및 가용 영역을 중심으로 구축됩니다. AWS 리전에서는 물리적으로 분리되고 격리된 다수의 가용 영역을 제공하며 이러한 가용 영역은 짧은 지연 시간, 높은 처리량 및 높은 중복성을 갖춘 네트워크에 연결되어 있습니다. 가용 영역을 사용하면 중단 없이 가용 영역 간에 자동으로 장애 조치가 이루어지는 애플리케이션 및 데이터베이스를 설계하고 운영할 수 있습니다. 가용 영역은 기존의 단일 또는 다중 데이터 센터 인프라보다 가용성, 내결함성, 확장성이 뛰어납니다.

AWS 리전 및 가용 영역에 대한 자세한 내용은 AWS 글로벌 인프라를 참조하십시오.

CloudFront 오리진 장애 조치

Amazon CloudFront는 AWS 글로벌 인프라 지원 외에도 데이터 복원성 요구 사항을 지원하는 오리진 장애 조치 기능을 제공합니다. CloudFront는 엣지 로케이션 또는 POP(상호 접속 위치)라고 하는 전세계 데이터 센터 네트워크를 통해 콘텐츠를 제공하는 글로벌 서비스입니다. 콘텐츠가 엣지 로케이션에 캐시되어 있지 않은 경우 CloudFront에서는 최종 버전의 콘텐츠 소스로 식별한 오리진에서 해당 콘텐츠를 검색합니다.

오리진 장애 조치로 CloudFront를 설정하면 특정 시나리오의 복원력을 향상시키고 가용성을 높일 수 있습니다. 시작하려면 CloudFront에 대한 기본 오리진을 지정할 오리진 그룹과 두 번째 오리진을 만듭니다. CloudFront에서는 기본 오리진이 특정 HTTP 상태 코드 실패 응답을 반환할 때 자동으로 두 번째 오리진으로 전환합니다. 자세한 내용은 CloudFront 오리진 장애 조치를 통한 고가용성 최적화 (p. 209) 단원을 참조하십시오.

Amazon CloudFront의 인프라 보안

관리형 서비스인 Amazon CloudFront는 Amazon Web Services: 보안 프로세스 개요 백서에 설명된 AWS 글로벌 네트워크 보안 절차로 보호됩니다.

AWS에서 게시한 API 호출을 사용하여 네트워크를 통해 CloudFront에 액세스합니다. 클라이언트가 TLS(전송 계층 보안) 1.0 이상을 지원해야 합니다. TLS 1.2 이상을 권장합니다. 클라이언트는 Ephemeral Diffie-Hellman(DHE) 또는 Elliptic Curve Ephemeral Diffie-Hellman(ECDHE)과 같은 PFS(전달 완전 보안, Perfect Forward Secrecy)가 포함된 암호 제품군도 지원해야 합니다. Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다. 또한 요청은 IAM 보안 주체와 관련된 보안 액세스 키 및 액세스 키 ID를 사용하여 서명해야 합니다. 또는 AWS Security Token Service(AWS STS)를 사용하여 임시 보안 자격 증명을 생성하여 요청에 서명할 수 있습니다.

할당량

CloudFront에는 다음 할당량(이전에는 제한으로 지칭)이 적용됩니다. Lambda@Edge에는 기본 CloudFront 할당량에 추가하여 적용되는 특정한 할당량도 있습니다.

주제

- 일반 할당량 (p. 447)
- 웹 배포의 일반 할당량 (p. 447)
- WebSocket 연결에 대한 할당량 (p. 448)
- 화이트리스트에 등록된 쿠키에 대한 할당량(웹 배포만 해당) (p. 448)
- 화이트리스트에 등록된 쿼리 문자열에 대한 할당량(웹 배포만 해당) (p. 449)
- 사용자 지정 헤더에 대한 할당량(웹 배포만 해당) (p. 449)
- SSL 인증서에 대한 할당량(웹 배포만 해당) (p. 449)
- 무효화에 대한 할당량 (p. 450)
- 필드 레벨 암호화에 대한 할당량 (p. 450)
- Lambda@Edge에 대한 할당량 (p. 450)
- 요청 제한 시간 (p. 452)
- RTMP 배포에 대한 할당량 (p. 452)

일반 할당량

엔터티	기본 할당량
배포당 데이터 전송 속도	40Gbps 더 높은 할당량 요청
배포당 초당 요청 수	100,000건 더 높은 할당량 요청
CloudFront 웹 또는 RTMP 배포에 추가할 수 있는 태그	50
배포당 제공할 수 있는 파일 수	할당량 없음
헤더와 쿼리 문자열을 포함한 요청의 최대 길이(본문 내용 제외)	20,480바이트
URL의 최대 길이	8,192바이트
신뢰할 수 있는 서명자의 활성 CloudFront 키 페어 자세한 내용은 서명된 URL 및 서명된 쿠키(신뢰할 수 있는 서명자)를 만들 수 있는 AWS 계정 지정 (p. 118) 단원을 참조하십시오.	2

웹 배포의 일반 할당량

엔터티	기본 할당량
AWS 계정당 웹 배포 수	200

엔터티	기본 할당량
자세한 내용은 배포 만들기 (p. 35) 단원을 참조하십시오.	더 높은 할당량 요청
HTTP GET, POST 및 PUT 요청의 최대 파일 크기	20GB
오리진별 응답 제한 시간 자세한 내용은 오리진 응답 제한 시간 (p. 42) 단원을 참조하십시오.	4-60초 더 높은 할당량 요청
파일 압축: CloudFront가 압축하는 파일 크기의 범위 자세한 내용은 압축된 파일 제공 (p. 87) 단원을 참조하십시오.	1,000 ~ 10,000,000바이트
배포당 대체 도메인 이름(CNAME) 수 자세한 내용은 대체 도메인 이름(CNAME)을 추가하여 파일에 대해 사용자 지정 URL 사용 (p. 66) 단원을 참조하십시오.	100 더 높은 할당량 요청
배포당 오리진 수	25 더 높은 할당량 요청
배포당 오리진 그룹 수	10 더 높은 할당량 요청
계정당 오리진 액세스 ID	100 더 높은 할당량 요청
배포당 캐시 동작 수	25 더 높은 할당량 요청

WebSocket 연결에 대한 할당량

엔터티	기본 할당량
오리진 응답 제한 시간(유휴 제한 시간)	10분 CloudFront가 지난 10분 사이에 오리진에서 클라이언트로 전송된 데이터를 전혀 감지하지 못한 경우, 유휴 상태로 간주되어 연결이 종료됩니다.

화이트리스트에 등록된 쿠키에 대한 할당량(웹 배포만 해당)

엔터티	기본 할당량
캐시 동작당 화이트리스트 지정 쿠키 수	10

엔터티	기본 할당량
자세한 내용은 쿠키를 기반으로 콘텐츠 캐싱 (p. 198) 단원을 참조하십시오.	더 높은 할당량 요청
화이트리스트 지정 쿠키 이름의 총 바이트 수(오리진에 모든 쿠키를 전달하도록 CloudFront를 구성하는 경우에는 적용되지 않음)	512에서 화이트리스트 지정 쿠키 수를 뺀 수

화이트리스트에 등록된 쿼리 문자열에 대한 할당량(웹 배포만 해당)

엔터티	기본 할당량
화이트리스트로 지정된 쿼리 문자열에서 최대 문자 수	128자
동일한 파라미터 내의 모든 화이트리스트로 지정된 쿼리 문자열에 대한 총 최대 문자 수	512자
캐시 동작당 화이트리스트 쿼리 문자열 수 자세한 내용은 쿼리 문자열 파라미터 기반의 콘텐츠 캐싱 (p. 196) 단원을 참조하십시오.	10 더 높은 할당량 요청

사용자 지정 헤더에 대한 할당량(웹 배포만 해당)

엔터티	기본 할당량
캐시 동작당 화이트리스트 지정 헤더 수 자세한 내용은 요청 헤더에 따라 콘텐츠 캐싱 (p. 200) 단원을 참조하십시오.	10 더 높은 할당량 요청
사용자 지정 헤더: 오리진 요청에 추가하도록 CloudFront를 구성할 수 있는 사용자 지정 헤더의 최대 수 자세한 내용은 사용자 지정 헤더를 오리진 요청에 추가 (p. 248) 단원을 참조하십시오.	이름/값 페어 10개 더 높은 할당량 요청
사용자 지정 헤더: 헤더 이름의 최대 길이	256자
사용자 지정 헤더: 헤더 값의 최대 길이	1,783자
사용자 지정 헤더: 헤더 값과 이름을 모두 결합한 최대 길이	10,240자

SSL 인증서에 대한 할당량(웹 배포만 해당)

엔터티	기본 할당량
전용 IP 주소를 사용하여 HTTPS 요청을 제공하는 경우 AWS 계정당 SSL 인증서 수(SNI를 사용하여 HTTPS 요청을 제공하는 경우 할당량 없음) 자세한 내용은 CloudFront에서 HTTPS 사용 (p. 92) 단원을 참조하십시오.	2 더 높은 할당량 요청

엔터티	기본 할당량
CloudFront 웹 배포와 연결할 수 있는 SSL 인증서	1

무효화에 대한 할당량

엔터티	기본 할당량
파일 무효화: 와일드카드 무효화를 제외하고 활성 무효화 요청에 허용된 파일의 최대 수 자세한 내용은 파일 무효화 (p. 80) 단원을 참조하십시오.	3,000
파일 무효화: 허용된 활성 와일드카드 무효화의 최대 수	15
파일 무효화: 하나의 와일드카드 무효화가 처리할 수 있는 파일의 최대 수	할당량 없음

필드 레벨 암호화에 대한 할당량

엔터티	기본 할당량
암호화할 필드의 최대 길이 자세한 내용은 필드 레벨 암호화 사용으로 민감한 데이터 보호 (p. 185) 단원을 참조하십시오.	16KB
필드 레벨 암호화 구성 시 요청 본문의 최대 필드 수	10
필드 레벨 암호화 구성 시 요청 본문의 최대 길이	1MB
하나의 AWS 계정과 연결될 수 있는 필드 레벨 암호화 구성의 최대 수	10
하나의 AWS 계정과 연결될 수 있는 필드 레벨 암호화 프로필의 최대 수	10
하나의 AWS 계정에 추가될 수 있는 퍼블릭 키의 최대 수	10
하나의 프로필에 지정되도록 암호화할 수 있는 최대 필드 수	10
필드 레벨 암호화 구성과 연결할 수 있는 CloudFront 배포의 최대 수	20
필드 레벨 암호화 구성에 포함될 수 있는 쿼리 인수 프로필 매핑의 최대 수	5

Lambda@Edge에 대한 할당량

이 섹션의 할당량은 Lambda@Edge에 적용됩니다. 이러한 할당량은 기본 CloudFront 및 Lambda 할당량에 추가하여 적용됩니다. 기본 할당량은 이 가이드의 할당량 (p. 447) 및 AWS Lambda Developer Guide의 할당량을 참조하십시오.

> **Note**
>
> Lambda는 계정 할당량 내에서 트래픽 증가에 대응하여 용량을 동적으로 확장합니다. 자세한 내용은 AWS Lambda Developer Guide의 함수 규모 조정을 참조하십시오.

추가로 Lambda@Edge 함수 사용 시 기타 제한 사항이 있습니다. 자세한 내용은 Lambda 함수에 대한 요구 사항 및 제한 사항 (p. 357) 단원을 참조하십시오.

이벤트 유형에 따라 다른 할당량

엔터티	오리진 요청 및 응답 이벤트 할당량	최종 사용자 요청 및 응답 이벤트 할당량
함수 메모리 크기	Lambda 할당량과 동일	128MB
함수 제한 시간. 이 함수는 AWS 리전에서 Amazon S3 버킷, DynamoDB 테이블 또는 Amazon EC2 인스턴스와 같은 리소스에 대한 네트워크 호출을 수행할 수 있습니다.	30초	5초
헤더 및 본문을 포함하여 Lambda 함수에서 생성되는 응답의 크기입니다.	1MB	40KB
Lambda 함수 및 포함된 라이브러리의 최대 압축 크기	50MB	1MB

기타 할당량

엔터티	기본 할당량
트리거를 생성할 수 있는 AWS 계정당 배포 수입니다.	25 더 높은 할당량 요청
배포당 트리거 수	100 더 높은 할당량 요청
초당 요청	10,000(각 리전에서) 더 높은 할당량 요청
동시 실행 자세한 내용은 AWS Lambda Developer Guide의 함수 규모 조정을 참조하십시오.	1,000(각 리전에서) 더 높은 할당량 요청

URI 및 쿼리 문자열의 크기 할당량

Lambda@Edge 함수에서 URI 또는 쿼리 문자열에 액세스하거나 업데이트할 때 URI의 총 길이는 쿼리 문자열을 포함하여 8,192자 미만이어야 합니다.

본문 포함 옵션이 적용된 요청 본문에 대한 크기 할당량

Include Body(본문 포함) 옵션을 선택해 Lambda@Edge 함수에 요청 본문을 노출하면 노출 또는 교체된 본문 부분에 다음 크기 할당량이 적용됩니다.

> **Note**
>
> 본문은 노출 전 항상 Lambda@Edge로 base64 인코딩됩니다.

본문을 Lambda 함수에 노출 시 크기 할당량

요청 본문이 크면 노출되기 전에 Lambda@Edge가 다음과 같이 자릅니다.

- 최종 사용자 요청의 경우 본문은 40KB에서 잘립니다.
- 오리진 요청의 경우 본문은 1MB에서 잘립니다.

Lambda 함수에서 요청 본문 반환 시 크기 할당량

요청 본문에 읽기 전용으로 액세스하는 경우 전체 원본 요청 본문이 오리진으로 반환됩니다. 그러나 요청 본문을 바꾸도록 선택한 경우 Lambda 함수에서 반환 시 다음 본문 크기 할당량이 적용됩니다.

본문 인코딩 유형	허용되는 본문 크기: 최종 사용자 요청	허용되는 본문 크기: 오리진 요청
text	40KB	1MB
base64	53.2KB	1.33MB

요청 제한 시간

엔터티	기본 할당량
요청 제한 시간 자세한 내용은 오리진 응답 제한 시간 (p. 243) 단원을 참조하십시오.	30초 더 높은 할당량 요청

RTMP 배포에 대한 할당량

엔터티	기본 할당량
AWS 계정당 RTMP 배포 수 자세한 내용은 RTMP 배포 작업 (p. 268) 단원을 참조하십시오.	100 더 높은 할당량 요청

Amazon CloudFront 관련 정보

CloudFront는 사용이 간편하면서도 다양한 기능을 제공합니다. 여기 나와 있는 정보 및 리소스는 CloudFront에 대해 자세히 알아보는 데 도움이 될 수 있습니다.

주제

- 추가 Amazon CloudFront 설명서 (p. 453)
- 지원 받기 (p. 453)
- CloudFront 개발자 도구 및 SDK (p. 453)
- Amazon Web Services 블로그에서 얻을 수 있는 팁 (p. 454)
- 객체 무효화 (p. 454)
- 비공개 콘텐츠 구성을 위한 도구 및 코드 예제 (p. 454)

추가 Amazon CloudFront 설명서

다음 표에는 이 서비스를 이용할 때 참조할 수 있는 관련 리소스가 나와 있습니다.

- Amazon CloudFront API Reference – API 작업, 파라미터, 데이터 형식에 대한 전체 설명과 서비스가 반환하는 오류 목록이 제공됩니다.
- CloudFront 새 소식 – 새로운 CloudFront 기능 및 최근에 추가된 엣지 로케이션 발표입니다.
- Amazon Simple Storage Service(S3)용 기술 문서 – 시작하기 기본, 서비스 개요, 프로그래밍 참조, API 참조 등 Amazon S3 서비스에 대한 세부적인 설명을 제공합니다.
- Amazon CloudFront 제품 정보 – 기능 및 요금 정보를 비롯해 CloudFront에 대한 정보를 얻을 수 있는 기본 웹 페이지입니다.

지원 받기

CloudFront에 대한 지원이 다양한 형태로 제공됩니다.

- 토론 포럼 – CloudFront 관련 기술적 질문을 논의할 수 있는 개발자를 위한 커뮤니티 기반 포럼입니다.
- AWS Support Center – 이 사이트에서는 최근 지원 사례에 대한 정보와 AWS Trusted Advisor 및 상태 확인의 결과를 함께 제공합니다. 또한 토론 포럼, 기술 FAQ, 서비스 상태 대시보드에 대한 링크와 AWS 지원 계획에 대한 정보를 제공합니다.
- AWS Premium Support 정보 – 1:1 신속 대응 지원 채널을 통해 AWS Infrastructure Services에서 애플리케이션을 구축하고 실행할 수 있도록 지원하는 AWS Premium Support 관련 정보의 기본 웹 페이지입니다.
- 문의처 – 결제 또는 계정과 관련하여 문의할 수 있는 링크입니다. 기술적인 질문의 경우 토론 포럼이나 위의 지원 링크를 이용하십시오.

CloudFront 개발자 도구 및 SDK

개발자 도구 페이지에서는 AWS로 혁신적인 애플리케이션을 구축하는 데 도움이 될 수 있는 설명서, 코드 샘플, 출시 정보 및 기타 정보를 제공하는 개발자 리소스 링크를 참조할 수 있습니다.

또한 Amazon Web Services에서는 CloudFront를 프로그래밍 방식으로 접근하기 위한 소프트웨어 개발 키트를 제공합니다. SDK 라이브러리는 서비스 요청에 암호화된 서명 적용, 요청 재시도, 오류 응답 처리를 포함한 여러 공통적인 작업을 자동화합니다.

- Java용 AWS SDK – 설정 및 기타 설명서
- .NET용 AWS SDK – 설정 및 기타 설명서
- PHP용 AWS SDK – 설정 및 기타 설명서
- Ruby용 AWS SDK – 설정 및 기타 설명서

Amazon Web Services 블로그에서 얻을 수 있는 팁

AWS 블로그에는 CloudFront 사용에 도움이 되는 여러 게시물이 올라와 있습니다. 예를 들어 CloudFront를 통한 Drupal 및 WordPress 사용에 관한 다음 블로그 게시물을 참조하십시오.

- Amazon CloudFront를 사용하여 Drupal 콘텐츠 가속화
- Amazon CloudFront를 사용한 WordPress 사이트 가속화 방법

객체 무효화

CloudFront에서 제공하는 무효화 방법에 추가로, 다음 타사 도구를 사용하여 객체를 무효화할 수 있습니다.

> **Note**
>
> 이러한 도구는 Amazon Web Services와는 관련이 없는 타사 공급업체가 개발한 것입니다. 이러한 도구의 사용 방법에 대한 자세한 내용은 해당 공급업체의 설명서를 참조하거나 이 공급업체에 문의하십시오.

- CloudBerry Explorer – http://cloudberrylab.com
- Ylastic – http://ylastic.com
- Cyberduck – http://cyberduck.io
- CloudFront Invalidator – https://github.com/swook/jQuery-CloudFront-Invalidator
- CDN Planet CloudFront Purge Tool – http://www.cdnplanet.com/tools/cloudfront-purge-tool/

Github(https://github.com)에서 코드 샘플을 검색할 수도 있습니다. CloudFront invalidation이라는 문구로 검색을 수행하십시오.

비공개 콘텐츠 구성을 위한 도구 및 코드 예제

CloudFront에서 제공하는 방식에 추가로, 다음 타사 도구에서는 비공개 콘텐츠의 배포를 구성하기 위한 웹 양식을 제공합니다. 일부 도구에서는 서명된 URL 생성을 위한 웹 양식도 제공합니다.

- CloudBerry – 비공개 콘텐츠에 대한 배포 구성과 서명된 URL의 생성을 지원합니다.

  CloudFront 비공개 콘텐츠에 CloudBerry를 사용하는 방법에 대한 자세한 내용은 How to Configure Private Content for CloudFront Streaming with CloudBerry를 참조하십시오.

  기본 루트 객체 설정에 대한 자세한 내용은 How to set CloudFront Default Object with CloudBerry S3 Explorer를 참조하십시오.

비공개 콘텐츠에 대한 자세한 내용은 AWS 블로그에서 블로그 게시물 New Amazon CloudFront Feature: Private Content를 참조하십시오.

사용자 지정 도메인 이름과 함께 서명된 쿠키를 사용하고, SSL/TLS 인증을 cloudfront.net 도메인에 사용하는 방법에 대한 자세한 내용은 Space Vatican 블로그 게시물 Using CloudFront Signed Cookies를 참조하십

시오. 이를 통해 대체 도메인 이름과 HTTPS 사용 (p. 102)에 설명된 대로 SNI(서버 이름 표시)를 지원하지 않는 최종 사용자(클라이언트)가 전용 IP 주소의 비용을 발생시키지 않고 HTTPS와 함께 대체 도메인 이름을 사용할 수 있습니다.

문서 기록

- API 버전: 2019-03-26
- 최종 설명서 업데이트: 2020년 2월 27일

변경 사항	설명	변경 날짜
Lambda@Edge에서 최신 런타임 버전 지원	이제 Lambda@Edge가 Node.js 12 및 Python 3.8 런타임을 사용하여 Lambda 함수를 지원합니다. 자세한 내용은 Lambda 기능 지원 실행 시간 및 구성 (p. 360) 단원을 참조하십시오.	2020년 2월 27일
CloudWatch의 새로운 실시간 지표	이제 Amazon CloudFront에서 Amazon CloudWatch에 실시간 지표 8개를 추가로 제공합니다. 자세한 내용은 추가 CloudFront 배포 지표 보기 (p. 417) 단원을 참조하십시오.	2019년 12월 19일
액세스 로그의 새 필드	CloudFront가 새로운 필드 7개를 액세스 로그에 추가합니다. 자세한 내용은 웹 배포 로그 파일 형식 (p. 399) 단원을 참조하십시오.	2019년 12월 12일
AWS for WordPress 플러그인	AWS for WordPress 플러그인을 사용하여 WordPress 웹 사이트 방문자에게 CloudFront를 통해 가속화된 시청 경험을 제공할 수 있습니다. 자세한 내용은 간단한 배포로 시작하기 (p. 22) 단원을 참조하십시오.	2019년 10월 30일
태그 기반 및 리소스 IAM 권한 정책	이제 CloudFront는 IAM 권한 정책을 지정하는 두 가지 추가 방법을 지원합니다. 이 두 가지 추가 방법은 태그 기반 및 리소스 수준 정책 권한입니다. 자세한 내용은 리소스에 대한 액세스 관리를 참조하십시오.	2019년 8월 8일
Python 프로그래밍 언어에 대한 지원	이제 Node.js 외에도 Python 프로그래밍 언어를 사용하여 Lambda@Edge에서 함수를 개발할 수 있습니다. 다양한 시나리오에 적용되는 예제 함수는 Lambda@Edge 예제 함수를 참조하십시오.	2019년 8월 1일
모니터링 그래프 업데이트	CloudFront 콘솔에서 직접 CloudFront 배포와 관련된 Lambda 함수를 모니터링하여 오류를 보다 쉽게 추적하고 디버깅할 수 있는 새로운 방법을 설명하는 내용이 업데이트되었습니다. 자세한 내용은 CloudFront 모니터링을 참조하십시오.	2019년 6월 20일
보안 콘텐츠 통합	새로운 보안 장에서는 CloudFront의 기능에 대한 정보와 데이터 보호, IAM, 로깅, 규정 준수 등의 구현에 대한 정보를 통합합니다. 자세한 내용은 보안 단원을 참조하십시오.	2019년 5월 24일
도메인 검증이 이제 필수	CloudFront는 이제 사용자가 SSL 인증서를 사용하여 배포에 대체 도메인 이름을 사용할 권한이 있는지 확인할 것을 요구합니다. 자세한 내용은 대체 도메인 이름 및 HTTPS 사용을 참조하십시오.	2019년 4월 9일
업데이트된 PDF 파일 이름	Amazon CloudFront 개발자 안내서의 새로운 파일 이름은 AmazonCloudFront_DevGuide입니다. 이전 이름은 cf-dg였습니다.	2019년 1월 7일
새로운 기능	CloudFront는 현재 클라이언트와 서버 사이에 수명이 긴 연결이 필요할 때 유용한 TCP 기반 프로토콜인 WebSocket을 지원합니다. 또한 고가용성을 필요로 하는 시나리오에 대한 오리진 장애 조치를 사용하여 이제 CloudFront를 설정할 수 있습니다. 자세한 내용은 CloudFront 배포를 통한 WebSocket 사용 및 CloudFront 오리진 장애 조치를 통한 고가용성 최적화 단원을 참조하십시오.	2018년 11월 20일

변경 사항	설명	변경 날짜
새로운 기능	CloudFront는 이제 Lambda 함수를 실행하는 HTTP 요청에 대한 상세 오류 로깅을 지원합니다. 로그를 CloudWatch에 저장하여 함수가 잘못된 응답을 반환할 경우 HTTP 5xx 오류 문제를 해결하는 데 사용할 수 있습니다. 자세한 내용은 Lambda 함수용 CloudWatch 지표 및 CloudWatch 로그 단원을 참조하십시오.	2012년 10월 8일
새로운 기능	이제 Lambda 함수에서 액세스할 수 있도록 Lambda@Edge가 본문에 쓰기 가능한 HTTP 메서드 (POST, PUT, DELETE 등)에 대한 요청을 노출하도록 선택할 수 있습니다. 읽기 전용 액세스 권한을 선택하거나 본문을 바꾸도록 지정할 수 있습니다. 자세한 내용은 본문 포함 옵션을 선택해 요청 본문에 액세스 단원을 참조하십시오.	2018년 8월 14일
새로운 기능	이제 CloudFront는 gzip에 추가로, 또는 gzip 대신에 brotli나 기타 압축 알고리즘으로 압축된 콘텐츠의 제공을 지원합니다. 자세한 내용은 압축 파일 서비스 단원을 참조하십시오.	2018년 7월 25일
재구성	Amazon CloudFront 개발자 안내서가 관련 콘텐츠 찾기를 간소화하고 검색 및 검색 기능을 향상하도록 재구성되었습니다.	2018년 28월 6일
새로운 기능	Lambda@Edge를 통해 이제 Amazon S3 버킷에 저장된 콘텐츠 전송을 사용자 지정할 수 있습니다. 이는 오리진 대상 이벤트 내에서 사용자 지정 헤더를 포함한 추가 화이트리스트 지정 헤더에 대한 액세스를 허용함으로써 가능합니다. 자세한 내용은 최종 사용자 위치 및 최종 사용자 디바이스 유형을 기반으로 한 콘텐츠 맞춤화를 보여주는 예제를 참조하십시오.	2018년 3월 20일
새로운 기능	이제 Amazon CloudFront를 사용하여 ECDSA(Elliptic Curve Digital Signature Algorithm)를 사용한 오리진에 대한 HTTPS 연결을 협상할 수 있습니다. ECDSA는 더욱 빠르고 이전 RSA 알고리즘처럼 안전한 크기가 작은 키를 사용합니다. 자세한 내용은 CloudFront와 오리진 간 통신에 지원되는 SSL/TLS 프로토콜 및 암호 및 RSA 및 ECDSA 암호 소개 단원을 참조하십시오.	2018년 3월 15일
새로운 기능	Lambda@Edge는 오리진의 오류 응답을 사용자 지정하는 기능을 제공합니다. 이제 Amazon CloudFront가 오리진으로부터 수신하는 HTTP 오류에 대해 Lambda 함수를 실행할 수 있습니다. 자세한 내용은 다른 위치로의 리디렉션 및 200 상태 코드(OK)와 함께 응답 생성을 보여주는 예제를 참조하십시오.	2017년 12월 21일
새로운 기능	새로운 CloudFront 기능인 필드 레벨 암호화는 주민등록번호와 같은 신용 카드 번호 또는 개인 식별 정보(PII) 등의 중요한 보안 데이터를 강화하는 데 도움이 됩니다. 자세한 내용은 필드 레벨 암호화 사용으로 민감한 데이터 보호 (p. 185) 단원을 참조하십시오.	2017년 12월 14일
문서 기록 아카이빙	이전 문서 기록이 아카이빙되었습니다.	2017년 12월

AWS Glossary

For the latest AWS terminology, see the AWS Glossary in the AWS General Reference.